TOWARD FASTER AND MORE ACCURATE STAR SENSORS USING RECURSIVE CENTROIDING AND STAR IDENTIFICATION

A Dissertation

by

MALAK ANEES SAMAAN

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

August 2003

Major Subject: Aerospace Engineering

TOWARD FASTER AND MORE ACCURATE STAR SENSORS USING RECURSIVE CENTROIDING AND STAR IDENTIFICATION

A Dissertation

by

MALAK ANEES SAMAAN

Submitted to Texas A&M University
in partial fulfillment of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

Approved as to style and content by:

John L. Junkins (Chair of Committee)	Daniele Mortari (Member)
Srinivas R. Vadali (Member)	Thomas C. Pollock (Member)
Michael Speed (Member)	Walter E. Haisler (Head of Department)

August 2003

Major Subject: Aerospace Engineering

ABSTRACT

Toward Faster and More Accurate Star Sensors
Using Recursive Centroiding and Star Identification. (August 2003)
Malak Anees Samaan, B.Sc., Cairo University;
M.S., Texas A&M University
Chair of Advisory Committee: Dr. John L. Junkins

The objective of this research is to study different novel developed techniques for spacecraft attitude determination methods using star tracker sensors. This dissertation addresses various issues on developing improved star tracker software, presents new approaches for better performance of star trackers, and considers applications to realize high precision attitude estimates.

Star-sensors are often included in a spacecraft attitude-system instrument suite, where high accuracy pointing capability is required. Novel methods for image processing, camera parameters ground calibration, autonomous star pattern recognition, and recursive star identification are researched and implemented to achieve high accuracy and a high frame rate star tracker that can be used for many space missions. This dissertation presents the methods and algorithms implemented for the one Field of View 'FOV' StarNavI sensor that was tested aboard the STS-107 mission in spring 2003 and the two fields of view StarNavII sensor for the EO-3 spacecraft scheduled for launch in 2007. The results of this research enable advances in spacecraft attitude determination based upon real time star sensing and pattern recognition. Building upon recent developments in image processing, pattern recognition algorithms, focal plane detectors, electro-optics, and microprocessors, the star tracker concept utilized in this research has the following key objectives for spacecraft of the future: lower

cost, lower mass and smaller volume, increased robustness to environment-induced aging and instrument response variations, increased adaptability and autonomy via recursive self-calibration and health-monitoring on-orbit. Many of these attributes are consequences of improved algorithms that are derived in this dissertation.

To the soul of my father, my mother, my wife and my kids

ACKNOWLEDGMENTS

First of all thanks to God.

I would like to express my deep thanks and gratitude to Dr. John L. Junkins for providing me the opportunity to pursue this dissertation topic. Thanks for spending time and effort to guide me and support me through the doctoral program; providing opportunities for attending conferences and extracurricular learning experiences. This work would not have been possible without his constant nurturing and assistance.

I would like thank Dr. Daniele Mortari for his support and advice and I am very fortunate to have worked with him in most of my research.

I also thank Drs. Srinivas Vadali and Tom Pollock for their support, suggestions and valuable help and comments.

A special thanks goes to Ms. Lisa Willingham for her administrative assistance. I extend my thanks to the entire administrative staff of the Department of Aerospace Engineering which ensured my smooth stay at Texas A&M.

I am indeed grateful to my friends, Wessam, Hussam and Sherif who stood by me and supported me in good and bad times. Thanks to my colleagues, Todd, Christian and Puneet for their help and support.

Finally, many thanks must go out to my family. I would like to thank my mother and my kids for their unconditional love and support. My love and appreciation go to my wife, Rania, for her devotion, encouragement, and confidence in me.

TABLE OF CONTENTS

CHAPTER Page

CHAPTER Page

LIST OF FIGURES

FIGURE Page

FIGURE Page

LIST OF TABLES

CHAPTER I

INTRODUCTION

1.1 Attitude Determination

Attitude determination is the process of estimating the orientation of a spacecraft by making remote observations of other celestial bodies or reference points [1]. Combinations of these sensor observations are used to generate a more accurate estimate of spacecraft attitude. Attitude estimates must be calculated quickly and continuously during the entire operational life of the spacecraft mission. During normal operations, the problem is recursive, the attitude filter based new predictions on present and prior sensor information [2]. The attitude filter must also estimate from activation when the spacecraft is first initiated and no prior data is available, i.e., the "lost-in-space" problem. Since spacecraft orientation will vary with time and task, orientation changes must be tracked, and rates and in some cases the accelerative torques estimated. The systems designed to carry out 3-axis attitude determination are inevitably complex, but must still be designed with the utmost care to perform the task as reliably as possible. Even temporary malfunctions are potentially serious, damaging fragile instruments, breaking communications links, upsetting measurements and disrupting power generation [3].

The journal model is *AIAA Journal of Guidance, Control and Dynamics.*

1.2 Survey of Spacecraft Attitude Determination Sensors

To determine the attitude and position of a spacecraft its orientation and location relative to some frame of reference of well known celestial body must be defined. To accomplish this one or more reference vectors should be chosen to which the spacecraft position and attitude can be referenced. The most commonly used vectors are unit vectors directed toward the Sun, the center of the Earth, a known star, or the magnetic field of the Earth. An attitude sensor is a mechanism which measures the orientation of these reference vectors in the spacecraft body frame of reference. By determining the orientation of two or more of these reference vectors relative to the spacecraft axes, the orientation of the spacecraft in space can be determined. Similarly, the position of a spacecraft can be determined from several reference vectors by triangulation. This section will describe some of the types of hardware used in space navigation.

1.2.1 Sun Sensor

Sun sensors are the most commonly used attitude sensors and are flown on almost every satellite. They are, of course, optical sensors. The brightness of the sun, which makes it easy to distinguish among other solar and stellar objects, and its distance from the Earth, which makes it appear as nearly a point source, among other factors, make the design of the sensor and its attitude determination algorithms simple and inexpensive. Sun sensors need very little power for operation. Since most missions require solar power and have sun-sensitive equipment which needs protection against sunlight or sun heat, sun sensors are crucial to almost all spacecraft. The wide range of sun sensor use has resulted in a variety of sensor types and designs. There are basically three types of sensors: analog sensors, sun presence sensors, and digital sensors [2].

1.2.2 Horizon Sensor

The primary means for directly determining the attitude of a spacecraft relative to the Earth are horizon sensors (otherwise known as Earth sensors). The attitude of spacecraft relative to Earth is one of the most important aspects of attitude determination and control, particularly for communications (TDRSS), weather (GOES), Earth resources (LANDSAT), etc. satellites. Earth cannot be treated as a point source like the Sun, especially for near Earth satellites for which Earth covers about 40% of the sky. Therefore, merely detecting the Earth is by no means sufficient for attitude determination, even for the most crude attitude determination requirements. Hence most sensors are designed to detect the Earth's horizon. Horizon sensors are infrared devices that detect the contrast of the cold of space and the warmth of the Earth (or other body). The difficulties encountered in horizon sensors include setting triggering thresholds to distinguish between the true or hard horizon and the edge of the atmosphere (or in the case of the Moon, cold Moon verses hot Moon threshold setting), and Sun rejection capability .

1.2.3 Magnetometers

Magnetometers are a common attitude sensor in attitude determination for the following reasons: they are vector sensors, i.e. they measure both the direction and magnitude of the magnetic field; they require low power for operation and are light weight; they are reliable (up to certain altitudes) and can operate over a wide range of temperatures; finally, they have no moving parts. However, since the magnetic field of the Earth is not precisely known, Magnetometers are not accurate. Due to a lack of complete knowledge of the magnetic field model, the predicted direction and magnitude of the field at the spacecraft's position are subject to errors. In particular,

for altitudes above 1000 km where magnetic field strength becomes small enough, errors become substantial (magnetic field strength is inversely proportional to the cube of the distance from the center of Earth) [2].

1.2.4 Star Sensors

A star sensor is a device which measures the direction of a star in the spacecraft coordinate system. By comparing these coordinates with known star directions stored in a star catalog, orientation of the spacecraft can be determined. Star trackers are the most accurate means of attitude determination with accuracies down to arc seconds. However, star sensors have traditionally been expensive, heavy, and require high power [4, 5]. Moreover, computer hardware/software to process their data is more complex and extensive than any other attitude sensor. Another drawback is that star sensors are subject to occultation and interference from the Sun, the Earth, and other bright light sources. But in spite of these drawbacks, their increasing accuracy and versatility has resulted in their use in a variety of spacecraft and experiments [6].

1.2.5 Gyroscopes

By definition, a gyroscope, or gyro, is any instrument which uses a rapidly spinning mass to sense and respond to changes in the inertial orientation of its spin axis. There are three types of attitude sensing gyros: rate gyro (RGs), rate integrating gyros (RIGs), and optical gyros. These sensors measure spacecraft orientation change. Recent advances have led to solid state and fiber-optic methods for making angular rate measurements.

1.2.6 Accelerometers

Accelerometers are devices which will measure acceleration and hence can be used to update the position and velocity of spacecraft. They come in two varieties: open and closed loop. Open loop accelerometers are based on a mechanical spring-mass-damper system. That is, when the mass experiences an acceleration, it deflects the spring. The distance the spring deflects is a measure of how much acceleration was felt. The system is damped to increase the accuracy and range of the accelerometer. The spring-mass-damper system forms the basis for all accelerometer theory, but the open loop mechanical system is not used on spacecraft due to their large weights and size [2].

1.3 The Objective of This Dissertation

Star trackers are widely used in spacecraft attitude determination because they produce higher accuracy attitude measurements than any other existing sensors. During recent decades, star trackers have been used for attitude determination on many spacecraft missions [7]. The proposed research will begin by evaluating the major algorithms that are used in the StarNavI project and that will be refined for the GIFTS (EO-3) mission. The one and two field of view (FOV) image processing and star centroiding, for the lost in space case and the predictive mode case, will be studied in detail. Also, the problem of the camera parameter's calibration will be studied, in which we could estimate the camera principal point offsets and focal length. The star identification algorithms for the lost in space case and the recursive mode case are also studied in detail with novel methods invented and implemented for each case. Finally, we will discuss some other important issues that can be involved in the whole system build like, the probability of miss-match of the star pattern match algorithm,

the problem of the image smear, and the case of the star identification when the camera parameters are not well calibrated.

1.4 Dissertation Organization

This dissertation mainly describes all the functions of the one and two field of view star sensors. Novel algorithms are introduced which were presented in many conferences and journals in the last 3 years, actually each chapter of this work is presented as a separate research paper. In this work, we can summarize the software functions of any star tracker as a five main modules:

- The Camera Control
- Star Image Processing
- Star Pattern Identification
- Attitude Estimation
- Telemetry Interface

All these software functions are shown in Fig. 1.1. The first part, covered in Chapter II of this dissertation, is concerned with the star image processing using a new technique "Predictive Centroiding". In this chapter we will discuss first the basic ideas of star centroiding and then our new algorithm for predictive centroiding.

In Chapter III, the problem of the ground calibration will be shown in detail with actual night sky tests. The main objective of the *Ground Calibration* algorithm is to find accurate calibrated values for the focal length f used for obtaining the CCD image and the offsets (x_o, y_o) used for mapping the CCD image coordinates into corresponding "measured" unit vectors toward each measured star.

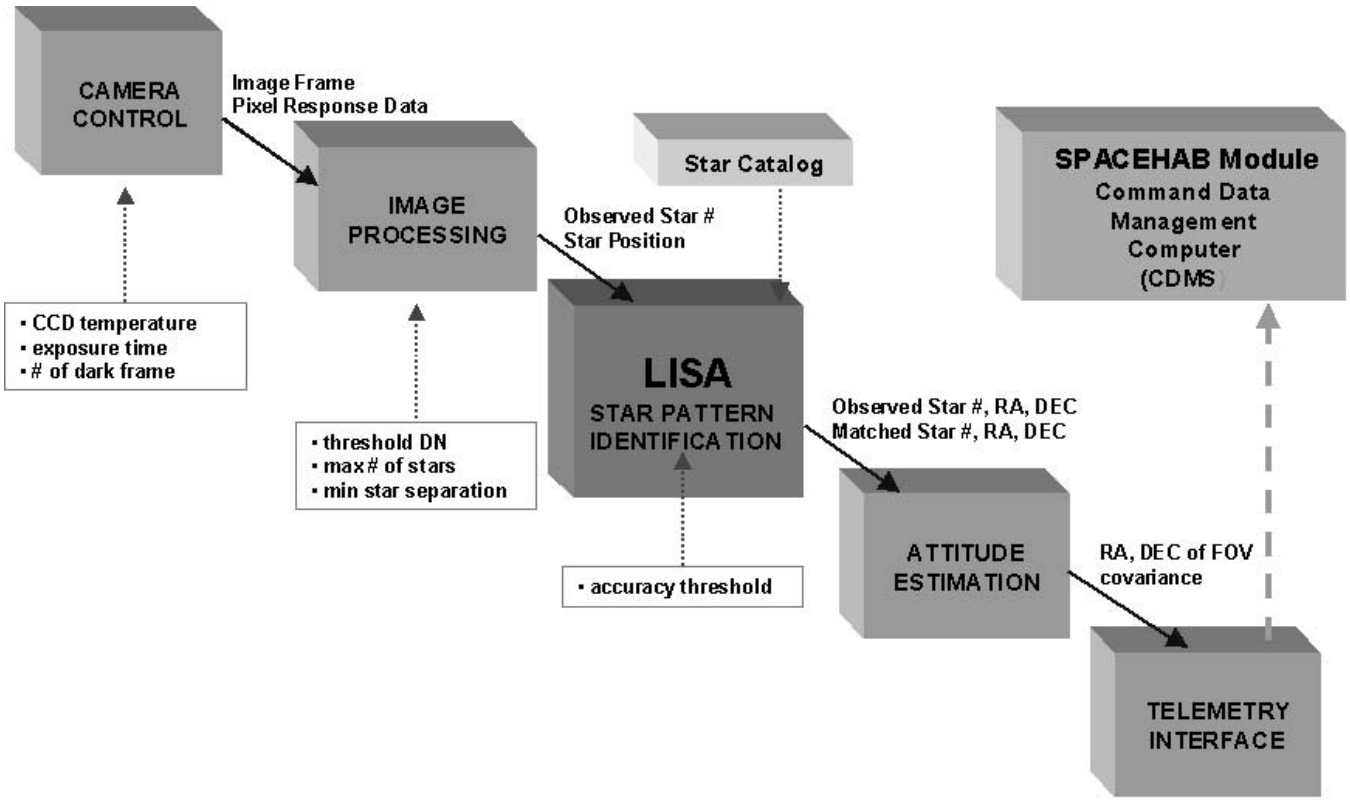

Figure 1.1 Functional block diagram of StarnavI flight software

Chapter IV, is concerned with the Lost-In-Space-Algorithm (LISA) and the details of the *Pyramid Algorithm*. This algorithm, based on the "Pyramid" structure of stars, is better than any other existing approach to solve the problem of star identification. We also show that the "Pyramid" is extremely efficient and robust.

Also, in Chapter V, we presented two new algorithms to identify the stars using the recursive mode star identification: these methods are ideally suited for the emerging active pixel camera technology. These algorithms are the *Star Neighbor Approach* and the *Spherical Polygon Search*.

In Chapter VI, we address the problem of the *Non-dimensional StarID*, in which the camera parameters are unknown or not well calibrated. A novel method will be presented which identifies stars independent of the scaling or location of triangles formed by sets of any three measured stars. These non-dimensional triangles are then matched with a catalog of triangles.

In Chapter VII, we will discuss the problem of the image smear in which the relatively high angular velocity of the spacecraft will affect the shape of the star in the star image.

Finally, in Chapter VIII, we will study the probability quantification of the star identification process. A novel analytical method to characterize the frequency of an incorrect star identification will be researched in this chapter.

1.5 Closing Remarks

In the following chapters, further attention is given to the detailed theoretical and experimental development of different algorithms of the star sensor. These algorithms are implemented to be used in the StarNavI project (flying in January 2003 on the space shuttle STS-107) and the GIFTS project (the EO-3 GIFTS mission scheduled

for 2007). All the algorithms in the following chapters are validated using night-sky tests and also Gaussian random images simulations. The numerical simulations of all algorithms were performed on a WINDOWS 98 based PC (650 MHz speed) using MATLAB and C/C++ languages.

CHAPTER II

PREDICTIVE CENTROIDING

2.1 Introduction

Star centroiding, locating the star center in a star image frame, is a fundamental process for any star tracker. In this chapter, the approximate locations of the stars in successive image frames are predicted using the angular velocity as provided by a rate gyro, then the centroid is updated based upon local image processing. When the rate gyro data are not available, then the angular velocity is estimated using the attitude kinematics equation and successive attitude estimates from the Lost-In-Space Algorithm. Also considered are the special features of non-circular star image shapes associated with optical tagging of starlight and/or image smear. Finally, an approach is presented to implement these ideas with the recently introduced Active Pixel Sensors, allowing dynamic pixel access and selected subarray analog-to-digital conversion of the pixel information is feasible, with logic dictated by the most recent image information and the instantaneous angular velocity estimate. This novel process that predicts the star image starlight locations is termed *predictive centroiding* (the contents of this chapter are summarized from [8] and [9]). These approaches, coupled with active pixel sensors, should enable near-optimal image processing and high frame rates. The contents of this chapter include analytical, computational, and night sky experimental results.

2.2 Basic Principles of Image Processing

The main objective of the image processing (centroiding) is to extract the key information and reduce a digitized image to a manageable data set that can be operated on and interpreted by a digital computer. In other words, the image processing is operates on the pixel response data to identify a visual pattern by using logic, which extracts the most reliable object features from the image obtained from a charge-coupled device (CCD) on active pixel sensor (APS). Figure 2.1 shows an actual night sky star image for Taurus and also a typical Point Spread Function (PSF) for starlight. In order to maximize the accuracy of the star location estimation, the starlight is usually de-focused over 3x3 to 15x15 pixel array masks, depending on the sensor CCD design and/or on the star magnitude. The resulting shape of the CCD starlight spot is usually described by point spread functions (PSFs), which are known to be near-Gaussian (bell-shaped). The appropriate optical designs defocus the starlight and utilizing modern image centroiding techniques allows estimating the star centroid with a precision of about 1/10 of a pixel or better. The image centroiding is, therefore, a fundamental process to increase the attitude data set accuracy since the centroid data forms the basis of observations. That is, accuracy of all subsequent processes is closely tied to the accuracy of the measurements. Recently, with the introduction of multiple Field of view star trackers [10] (which output non-circular images to identify the associated field of view), the centroiding algorithms have new challenges to meet. The speed and accuracy to accomplish the centroiding process represent the performance measures for comparison with existing centroiding approaches.

For the case of two superimposed orthogonal FOVs of a split field of view camera, figure 2.2 shows a two FOV star image. In Fig. 2.2, the elliptical star images associated with the "optical tagging method" for denoting which FOV the stars were imaged

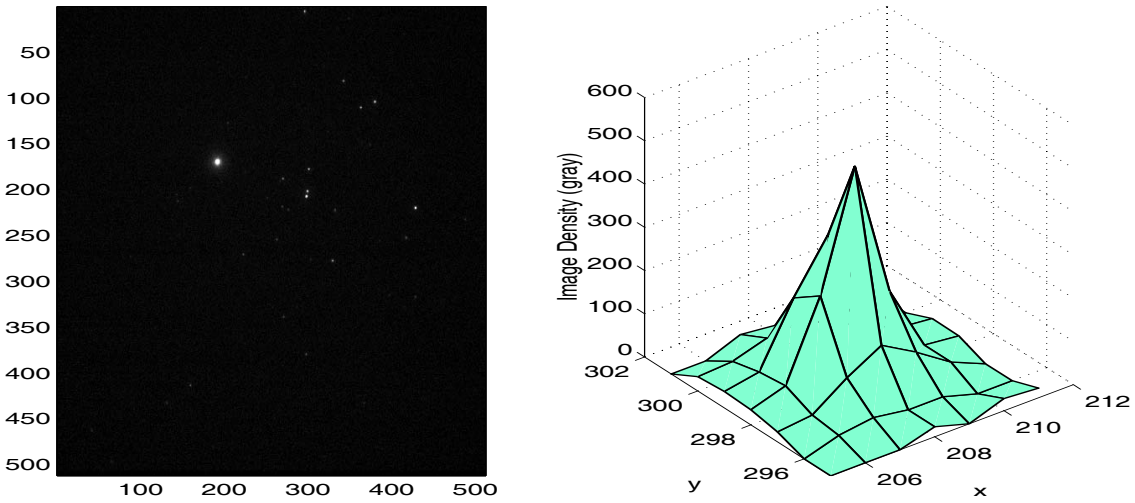

Figure 2.1 An actual star image and PSF response for a typical star

are shown. Notice that astigmatism distortion is deliberately introduced [10] to cause the normally circular PSF to become elliptical. The eigenvalue of the figure inertia tensor associated with the "stretched" direction (eigenvector), indicates the FOV of origin for each star image. For example, in Fig. 2.2, the vertically stretched elliptical images are from the left FOV, whereas the horizontally stretched images are from the right FOV. Obviously, the shape of the images can easily be detected during image processing by using the eigenvalue ratio between the inertia principal axes of the mask surrounding each non-circular star image.

2.3 Image Processing Technique

Because the defocused image energy distributions are actually surfaces, we therefore must define what is meant by image location. There are several ways to approach this problem. One possible method is to use a peak location algorithm, the most obvious peak locator would simply be a search for the pixel with the maximum gray scale value. This method would, however, defeat the purpose of defocusing the PSF

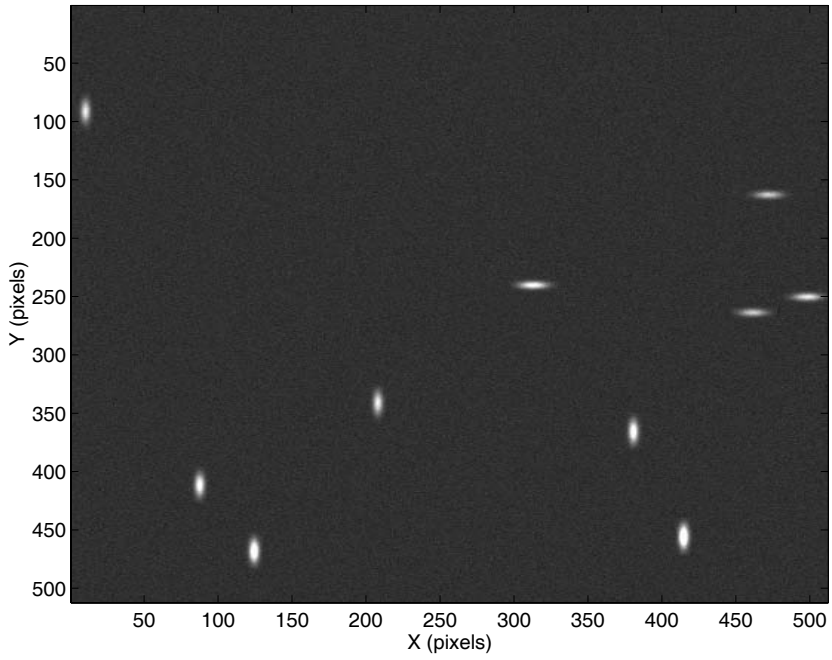

Figure 2.2 Two fields of view star images with deliberated astigmatism

in that the resolution of the focal plane image location would still be about one pixel [11]. Another method, called the moment method, involves the calculation of the star location in a manner similar to calculating the center of mass using the moment technique. Fig. 2.3 shows a magnification of a window surrounding star and the intensity value of each pixels.

The first moment method of image centroiding is adopted in previous research to detect the centroid of the brightness of the image, and is written as;

$$\hat{x} = x_m + \frac{\sum_{i=1}^{n}\sum_{j=1}^{n} x_{ij} I_{ij}}{\sum_{i=1}^{n}\sum_{j=1}^{n} I_{ij}} \quad \hat{y} = y_m + \frac{\sum_{i=1}^{n}\sum_{j=1}^{n} y_{ij} I_{ij}}{\sum_{i=1}^{n}\sum_{j=1}^{n} I_{ij}} \tag{2.1}$$

where;

(x_m, y_m) = Centroid location of the star with high pixel intensity,

I_{ij} = intensity (gray scale digital) value at the $(i,j)^{th}$ pixel,

(x_{ij}, y_{ij}) = position of the $(i,j)^{th}$ pixel,

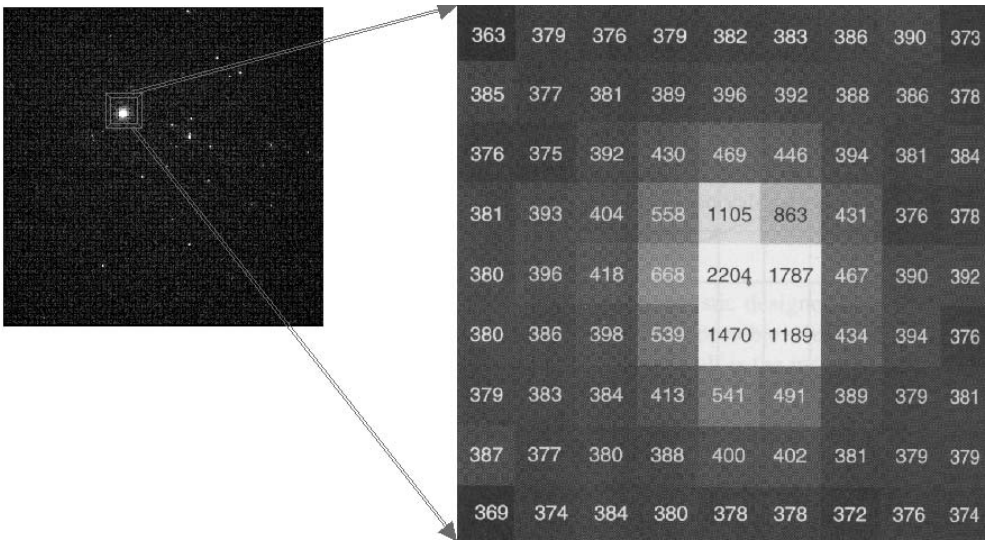

Figure 2.3 The intensity of the window surrounding the star

n= Size of the window used to compute the centroid, the size of the centroid window determined using peak intensity value, its value is normally between (3 x 3 to 13 x 13) depend on the peak value.

Equation (2.1) is also known as the simple *centroiding technique*. It can be shown that, at least when the PSF subtends only about three pixels, the first moment technique consistently estimates the centroid location [12].

Using the fact that the moment arm (x_{ij}) is constant along the columns of the pixel window and the moment arm (y_{ij}) is constant along the rows of the pixel window. Equation (2.1) could be written in another form, such that

$$\hat{x} = x_m + \frac{\sum_{k=0}^{m} (m-k)\,(C(n-k) - C(k+1))}{\sum_{i=1}^{n} C(i)} \tag{2.2}$$

$$\hat{y} = y_m + \frac{\sum_{k=0}^{m} (m-k)\,(R(n-k) - R(k+1))}{\sum_{i=1}^{n} R(i)} \tag{2.3}$$

where;

$$m = \frac{n-1}{2},\ C(k) = \sum_{j=1}^{n} I_{kj} \text{ and } R(k) = \sum_{i=1}^{n} I_{ik}$$

After locating the first peak (star), the one that has maximum intensity, the value of the pixels in the window of this peak are set to zero, to prevent this image from being found again. The same centroiding procedure is done recursively to each one of the image peaks to find all the stars in the image, which have an intensity value larger than the thresholding value.

2.4 Active Pixel Sensor (APS)

A new photon-sensitive imaging array, known as the active pixel sensor (APS) [13] has emerged as a competitor to the CCD imager for use in star and target tracking. The APS is based on the same complementary metal oxide semiconductor (CMOS) technology that enables inexpensive computer memory mass production. An active pixel sensor is defined as a detector array technology that has at least one active transistor within each pixel unit cell. That is, both photodetector and readout amplifier are integrated within each pixel. The APS architecture thus eliminates the need for many nearly perfect charge transfers. With the active pixel, the signal is driven from the pixel over X-Y metallic wires rather than being physically transported via a shift register. Since APS technology enables intra-pixel transfer instead of repetitive pixel-to-pixel charge transfers as in CCDs, APS imagers can be expected to be more radiation hard, operate well at lower temperatures, be fabricated more inexpensively in large array sizes, and be more compatible with advanced materials [14]. The inherent characteristics of APS are expected to reduce the random centroid error noise due to proton induced displacement damage and charge trapping. Since the APS has been developed using CMOS technology that has already achieved nearly the same

performance as a CCD image sensor, the use of CMOS technology permits ready integration of on-chip timing and control electronics, as well as signal chain electronics. Due to the inclusion of an on-chip analog-to-digital converters, direct communication with an onboard computer as well as the possible inclusion of on-chip centroiding algorithms suggests future trackers based on APS can significantly decrease power consumption. This feature is advantageous to the design of autonomous star trackers for future micro spacecraft where power consumption and mass are critical. Due to the independent nature of the APS readout from individual pixels, image blooming (generally present in CCDs when bright objects are present in the FOV) is negligible with APS. This means the dynamic range of star visual magnitude can be highly extended so that star trackers will operate better in the presence of bright objects in the FOV. This feature will significantly increase the sky coverage especially for earth orbiting star trackers. One of the most important characteristics of the APS is direct addressing of each pixel. That is, APS allows high speed random access to each pixel of the array with the improvement of the signal-to-noise ratio. A pixel is addressed in much the same way as a complete memory location. This allows the computer to read out local blocks of pixels (containing star images) in a high speed repetitive process, without disturbing or accessing pixels outside the window of interest, and this feature enables efficient tracking mode once the initial attitude acquisition has been accomplished. Thus only a fraction of pixels need to be accessed in the track mode, and the integration time can, in principly be customized (bright stars $\Leftrightarrow$ short integration time, faint stars $\Leftrightarrow$ large integration time), giving rise to highly adaptive star trackers that virtually never encounter "holes" in the sky where stars cannot be imaged. With the present state of the APS technology, one significant disadvantage has emerged. Since each pixel is co-located with additional solid state elements (amplifier and readout circuitry), a smaller fraction of the area is actually light sensitive.

Whereas the CCD detectors have "fill factors" with > 70 % of the area light sensitive, the APS detectors available have fill factors around 50 %. Even though only breadboard level of APS-based star trackers has been developed up to present, it is already evident that the technology will rapidly replace CCD star trackers in the near future due to the outstanding advantages of APS over CCD technology [15]. It is expected that APS fill factors will eventually compete with CCDs and then it is anticipated that the APS will become the detector of choice for most space applications.

2.5 Predictive Centroiding

The fact that the frame rate of star trackers using an advanced Active Pixel Sensor can be relatively high (10 to 100 Hz), and for small optics and associated realistic integration times, the maximum angular velocity will necessarily be fairly low (significantly less than one degree/sec). In view of these considerations, a typical star is imaged many successive times (typically, several hundred times) before it leaves the field of view. In successive image processing, to enable local access and analog-to-digital conversion of only those pixels where starlight is likely to be found, we can make use of previous star locations and the approximately linear displacement of the star locations from one frame to another one.

In many applications, we can simply assume $\omega\delta t$ is the differential angular displacement of the star sensor, about the instantaneous angular velocity vector, and map this rotation into linearly predicted displacements of the image centers for all stars, where ω is the angular velocity vector and δt is the time interval between successive frames. This linear approximation is usually more than adequate to locate the center of the starting adaptive mask for computing star centroids. Using the approach we present herein, there is negligible effect on the final centroid approxima-

Figure 2.4 StarnavII split field of view star camera

tion. The predictive centroiding algorithm is implemented to be used in the GIFTS EO-3 (Geostationary Imaging Fourier Transform Spectrometer) mission. Figure 2.4 provides a schematic of the split FOV camera that will be used in the GIFTS mission. The electro-optical details of this camera design are the subject of a pending patent [10].

2.5.1 Predictive Centroiding Steps

Once we have the star image, taken by a one or multiple FOVs camera, the centroiding process is needed. Initially, the image is treated using the usual centroiding techniques which use the mass moment method to find the location of the star in the image. This is the most common approach and has been motivated by the acquisition and tracking algorithms developed historically for the ASTROS star tracker developed at NASA JPL [16]. After the first image, the following images are processed using our new technique, *predictive centroiding*. Predictive centroiding is treated for the one, two and three Field Of View (FOV) star trackers [10, 17]. The steps of predictive

centroiding are stated as follows:

1. The attitude matrix (projecting body frame directions onto the inertial frame), $C(t_0)$ evaluated at initial time, is calculated using the Lost In Space Algorithm (LISA) [18].

2. Using the initial angular velocity data, we can predict the attitude matrix at the current frame $C_p(t+\delta t)$ using the previous attitude matrix $C(t)$, and the following linear approximation

$$C_p(t+\delta t) = [I - \tilde{\omega}\delta t]C(t) \tag{2.4}$$

   where $\tilde{\omega}$ is the cross product matrix populated with the components of the angular velocity vector ω. For virtually all current anticipated missions, $\|\tilde{\omega}\delta t\| < 10^{-3}$ rad., so Eq. (2.4) should be accurate to micro radian or better precision for one time step prediction.

3. The vectors associated with the four corners of the FOV are projected to the inertial reference frame using the predicted attitude matrix.

4. By accessing the star catalog using a bounding box whose vertices are the sensor four corners, we can access the inertial reference vectors $\hat{\mathbf{v}}_i$ to stars imaged in that frame.

5. Given the attitude matrix $C^{\mathrm{T}}(t) \equiv [\,\hat{\mathbf{c}}_1 \quad \hat{\mathbf{c}}_2 \quad \hat{\mathbf{c}}_3\,]$ at time t, and the inertial star vectors $\hat{\mathbf{v}}_i$, $(i = 1, \ldots, n)$, the star locations $(x_i,\, y_i)$ are evaluated by the co-linearity equations:

$$x_i = x_0 - f\,\frac{\hat{\mathbf{c}}_1^{\mathrm{T}}\hat{\mathbf{v}}_i}{\hat{\mathbf{c}}_3^{\mathrm{T}}\hat{\mathbf{v}}_i} \qquad \text{and} \qquad y_i = y_0 - f\,\frac{\hat{\mathbf{c}}_2^{\mathrm{T}}\hat{\mathbf{v}}_i}{\hat{\mathbf{c}}_3^{\mathrm{T}}\hat{\mathbf{v}}_i} \tag{2.5}$$

   where f is the camera focal length and x_0, y_0 are the optical axis offsets.

6. For the considered frame, once the star locations are predicted, then these locations become the center of masks used for centroiding for the real CCD image.

7. The recursive star identification algorithm [19], which uses the star neighbor approach, is then be used to identify the observed stars for that frame. The optimal estimate of the attitude matrix $C(t+\delta t)$ is determined by using the ESOQ-2 method [20], and the associated angular velocity estimate is calculated from kinematics equation $\dfrac{dC}{dt} = -\tilde{\omega}\ C$, by replacing differentials with small finite differences

$$\tilde{\omega} = \left[I - C(t+\delta t)\, C^{\mathrm{T}}(t)\right] / (\delta t) \tag{2.6}$$

8. The best estimate of angular velocity is used to determine the predicted star locations at next frame and the loop start again from step 2.

2.5.2 Simulation Results

Some simulation results are shown in Figs. 2.5 and 2.6, for a one FOV camera where two sequential images are superimposed and the two sets of star locations are shown to indicate the image motion. Also, for the case of two superimposed orthogonal FOVs of a split field of view camera (StarNav II), the Figs. 2.7 and 2.8 show the simulated star images and the star locations for two successive images.

In Fig. 2.7, the elliptical star images associated with the "Optical tagging method" for denoting which FOV the stars were imaged, are shown. Notice that astigmatism distortion is deliberately introduced [10] to cause the normally circular PSF to become elliptical. The eigenvalue of the figure inertia tensor associated with the "stretched" direction (eigenvector), indicates the FOV of origin for each star image. For example, in Fig. 2.7, the vertically stretched elliptical images are from the

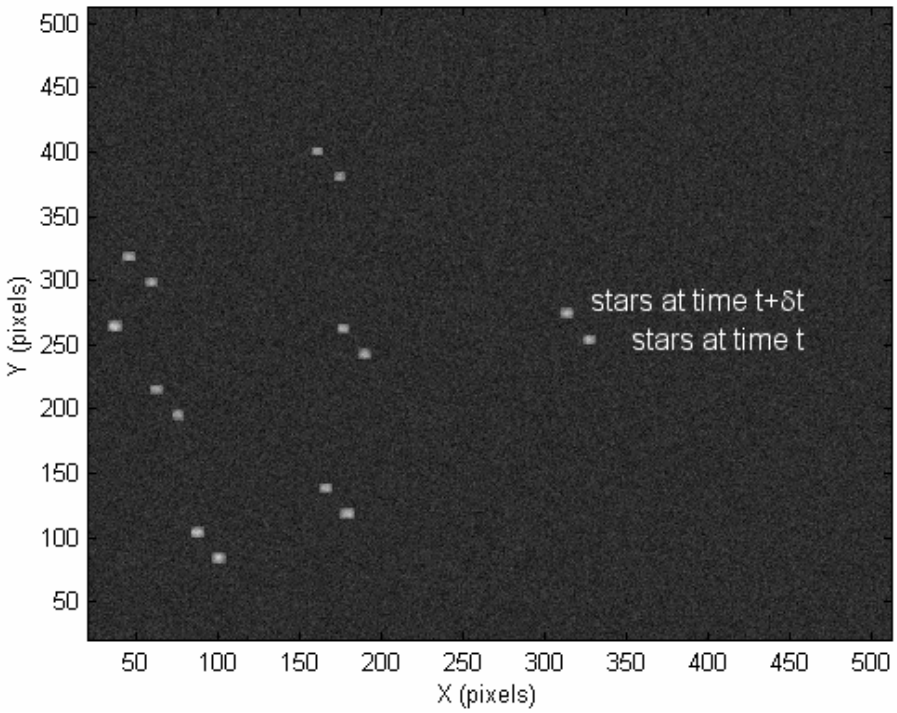

Figure 2.5 Two superimposed images for one FOV

left FOV, whereas the horizontally stretched images are from the right FOV. Obviously, the shape of the images can easily be detected during image processing by using the eigenvalue ratio between the inertia principal axes of the mask surrounding each non-circular star image. Alternatively, we can simply compute the second statistical moments in lieu of finding the eigenvalues of the covariance matrices. These results also show that the time required to find the centroids using predictive centroiding is less by one order of magnitude than the time required using the regular centroiding techniques. From a mathematical point of view, if $[x_c, y_c]$ are the coordinates of the centroiding, then

$$J_1 = \sum_i (y_i - y_c)^2 \qquad \text{and} \qquad J_2 = \sum_i (x_i - x_c)^2 \tag{2.7}$$

are approximately the principal figure moments of inertia, where the sums are extended to all of the pixels (coordinate $[x_i, y_i]$) used for centroiding. Note, by design,

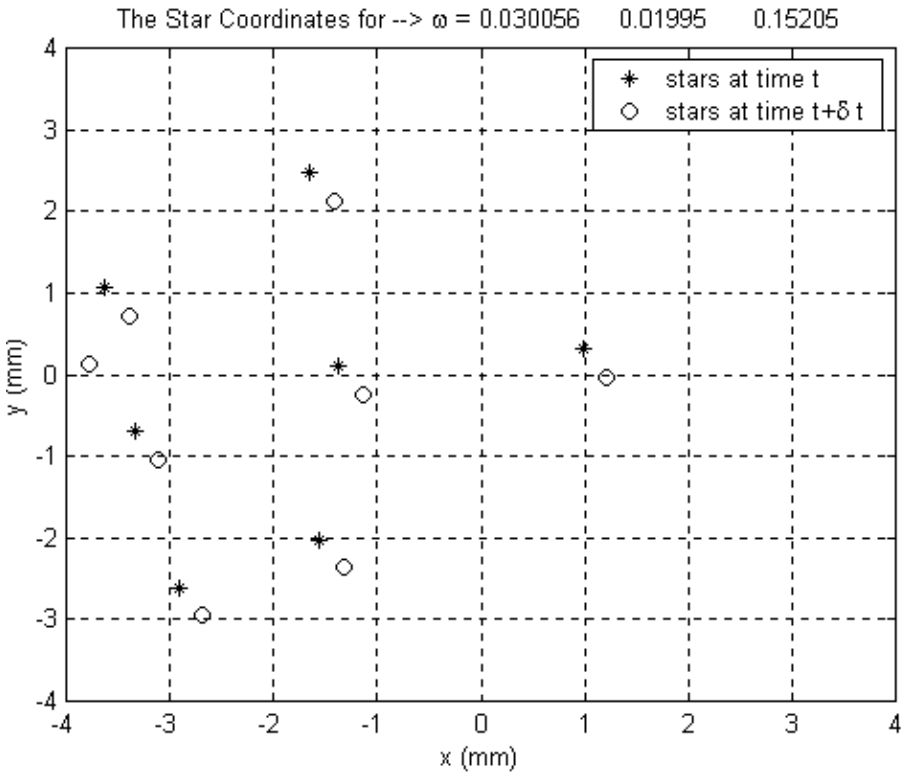

Figure 2.6 Single FOV image centroid locations

the (x_i, y_i) axes are the nominal principal axes of the astigmatic elliptical stars. Therefore, the FOV identification is simply dictated by

$$J_1 < J_2 \qquad \text{or} \qquad J_1 > J_2 \tag{2.8}$$

provided that $(J_1 - J_2)^2$ is greater than a given numerical threshold.

The number of pixels processed is reduced from $512 \cdot 512 = 262,144$ to approximately $(n \cdot m)$, where n is the number of stars and m is the number of pixels associated with the average star images. For a typical images, $n \cdot m \simeq 8 \cdot 25 \simeq 200$, so four order of magnitude less image data is involved; this suggests that further algorithm optimization may greatly improve over the one order of magnitude advantage gained.

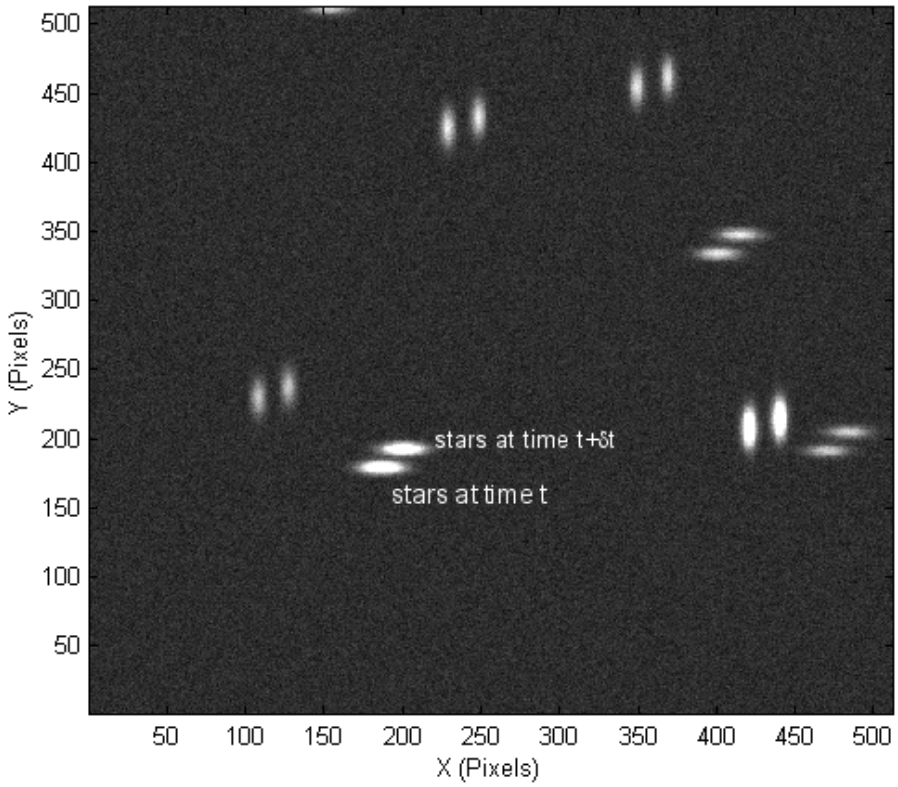

Figure 2.7 Two superimposed images for two FOVs

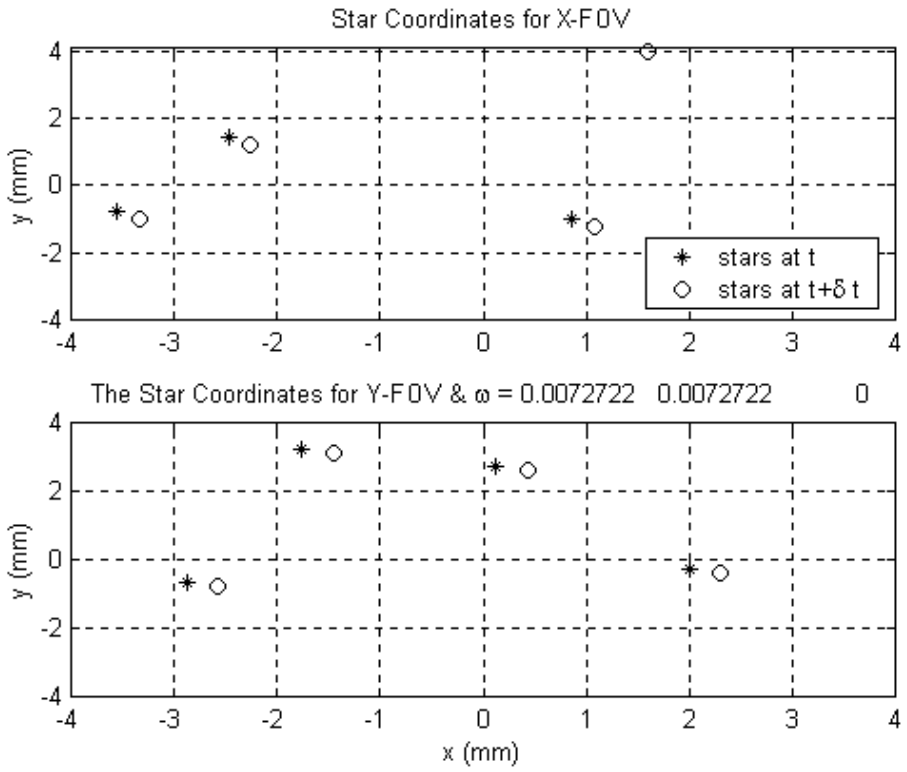

Figure 2.8 Two FOVs image centroid locations for the pair of images

CHAPTER III

GROUND CALIBRATION FOR THE BORE-SIGHT OFFSETS AND THE FOCAL LENGTH

3.1 Introduction

An important problem in spacecraft autonomy is the frequently occurring situation that a previously calibrated instrument encounters unexpected and unpredictable changes. For star trackers, recent missions, e.g. MSX [16] have shown the consequence that attitude estimation precision is degraded as a consequence.

In this chapter, we show a novel method for solving this problem (the contents of this chapter are summarized from [21]). The method makes use of residuals between measured interstar cosine angles and the interstar cosine angles between the corresponding cataloged stars ($\hat{\mathbf{v}}_i^T \hat{\mathbf{v}}_j = \hat{\mathbf{w}}_i^T \hat{\mathbf{w}}_j$, for perfect measurements) to learn the calibration corrections on-orbit, starting from ground-based calibration results.

Our approach makes use of the truth that interstar angles are an invariant of rotational transformations, and therefore we do not require knowledge of the generally unknown spacecraft attitude to estimate calibration parameters that offset interstar angles [22]. In this chapter the analytical and numerical solutions for the calibration of the principal point offset (x_o, y_o) and focal length (f) are developed.

3.2 The Ground (Batch) Calibration

The main objective of the *ground calibration* algorithms is to find accurate calibrated values for the focal length (f) used for obtaining the CCD and offset (x_o, y_o) used for mapping the CCD image coordinates into corresponding "measured" unit vectors

toward each measured star [21].

Without the focal plane calibration the problem of the Star Identification [18-23] may not give an accurate results due to the fact the the imaged stars have to be modified by the amount of the bore-sight offsets and the focal length used to get the corresponding unit vectors. The method of least squares optimal estimation has proved to be an extremely powerful technique, and has been used extensively in the parameter identification algorithms (Calibration). Our approach makes use of the truth that interstar angles are an invariant of rotational transformations, and therefore we do not require knowledge of the generally unknown spacecraft attitude to estimate calibration parameters that offset interstar angles. To begin the ground calibration algorithm, the stars of the CCD image have to be manually identified, or identified by some other highly robust algorithm tolerant of calibration errors [24], so that the inertial and the cataloged vectors are given as an input to the algorithm.

From the centroiding algorithm we have x_i and y_i for $(i = 1, \ldots, n)$ where n is the number of stars in the FOV. We can use an initial estimate of the focal length to generate a simulated set of unit vectors to the measured stars, with focal plane components of the vectors as

$$\hat{\mathbf{w}}_i = \frac{1}{\sqrt{((x_i - x_o)^2 + (y_i - y_o)^2 + f^2)}} \begin{pmatrix} -(x_i - x_o) \\ -(y_i - y_o) \\ f \end{pmatrix} \tag{3.1}$$

where f is the camera focal length, $(x_o\ ,\ y_o)$ are the bore-sight errors. Equation (3.1) represents the measurement model.

So, f , x_o and y_o are the unknowns to be estimated using nonlinear least square optimal estimation. We note that the length unit of (x, y, f, x_o, y_o) are arbitrary, any convenient choice (mm) or "pixels" can be used; the known focal plane

size provides the scale factor needed. If this scale factor is slightly in error, it will result in compensating converged offsets in (f, x_o, y_o) and the remaining calibration parameters.

3.2.1 The Cataloged Vectors

The inertial cataloged star direction cosines are calculated using;

$$\hat{\mathbf{v}}_i = \begin{pmatrix} \cos\alpha_i \cos\delta_i \\ \sin\alpha_i \cos\delta_i \\ \sin\delta_i \end{pmatrix} \tag{3.2}$$

Where α_i and δ_i are the right ascension and declination for star i ($i = 1, \ldots, n$) respectively. Now, to solve for the unknowns we can use the fact that the interstar angles for the perfectly imaged vectors and the cataloged vectors have to be the same, mathematically;

$$\hat{\mathbf{v}}_i^{\mathrm{T}} \hat{\mathbf{v}}_j = \hat{\mathbf{w}}_i^{\mathrm{T}} \hat{\mathbf{w}}_j \tag{3.3}$$

Now, using equation (3.1) we can show that;

$$\hat{\mathbf{v}}_i^{\mathrm{T}} \hat{\mathbf{v}}_j = \frac{N}{D_1 D_2} = g_{ij}(x_o, y_o, f) \tag{3.4}$$

Where

$$\begin{cases} N = (x_i - x_o)(x_j - x_o) + (y_i - y_o)(y_j - y_o) + f^2 \\ D_1 = \sqrt{(x_i - x_o)^2 + (y_i - y_o)^2 + f^2} \\ D_2 = \sqrt{(x_j - x_o)^2 + (y_j - y_o)^2 + f^2} \end{cases} \tag{3.5}$$

By using the linearization about the nominal value ($\hat{x}_o$,$\hat{y}_o$,$\hat{f}$) we have

$$x_o = \hat{x}_o + \Delta x, y_o = \hat{y}_o + \Delta y \text{ and } f = \hat{f} + \Delta f \tag{3.6}$$

Substitute equation (3.6) in (3.4) to get

$$\hat{\mathbf{v}}_i^{\mathrm{T}}\hat{\mathbf{v}}_j = g_{ij}(\hat{x}_o, \hat{y}_o, \hat{f}) + \begin{bmatrix} \frac{\partial g_{ij}}{\partial x_o} & \frac{\partial g_{ij}}{\partial y_o} & \frac{\partial g_{ij}}{\partial f} \end{bmatrix} \begin{Bmatrix} \Delta x_o \\ \Delta y_o \\ \Delta f \end{Bmatrix} \tag{3.7}$$

Let

$$R_{ij} = \hat{\mathbf{v}}_i^{\mathrm{T}}\hat{\mathbf{v}}_j - g_{ij}(\hat{x}_o, \hat{y}_o, \hat{f}) = \begin{bmatrix} \frac{\partial g_{ij}}{\partial x_o} & \frac{\partial g_{ij}}{\partial y_o} & \frac{\partial g_{ij}}{\partial f} \end{bmatrix} \begin{Bmatrix} \Delta x_o \\ \Delta y_o \\ \Delta f \end{Bmatrix} \tag{3.8}$$

For $(i = 1, \ldots, n-1)$, $(j = i+1, \ldots, n)$ and $(j \neq i)$ we can write equation (3.8) as

$$\{R\} = [A]\{\Delta Z\} \tag{3.9}$$

Where

$$A = \begin{bmatrix} \frac{\partial g_{12}}{\partial x_o} & \frac{\partial g_{12}}{\partial y_o} & \frac{\partial g_{12}}{\partial f} \\ \frac{\partial g_{13}}{\partial x_o} & \frac{\partial g_{13}}{\partial y_o} & \frac{\partial g_{13}}{\partial f} \\ . & . & . \\ . & . & . \\ \frac{\partial g_{n-1,n}}{\partial x_o} & \frac{\partial g_{n-1,n}}{\partial y_o} & \frac{\partial g_{n-1,n}}{\partial f} \end{bmatrix}, R = \begin{Bmatrix} R_{12} \\ R_{13} \\ . \\ . \\ R_{n-1,n} \end{Bmatrix} \text{ and } \Delta Z = \begin{Bmatrix} \Delta x_o \\ \Delta y_o \\ \Delta f \end{Bmatrix} \tag{3.10}$$

3.2.2 Least Squares Optimal Estimation

The method of least squares is a powerful and widely applied tool from estimation theory. There are many excellent sources on the subject where derivations of the equations can be found in [25], [26]. By using the least squares we can show that the solution ΔZ which minimize the residual of equation (3.9) in the least squares sense is given by;

$$\{\Delta Z\}_k = [A_k^T A_k]^{-1} A_k^T \{R\}_k \tag{3.11}$$

where $k = 1, 2, \ldots,$ No. of iterations

$$\begin{Bmatrix} \hat{x}_o \\ \hat{y}_o \\ \hat{f} \end{Bmatrix}_{k+1} = \begin{Bmatrix} \hat{x}_o \\ \hat{y}_o \\ \hat{f} \end{Bmatrix}_k + \begin{Bmatrix} \Delta x_o \\ \Delta y_o \\ \Delta f \end{Bmatrix}_k \tag{3.12}$$

The Jacobian partial derivative terms in equations (3.7), (3.8) are given by;

$$\frac{\partial g_{ij}}{\partial x_o} = \frac{D_1 D_2 (2x_o - x_i - x_j) + N[(x_i - x_o) D_2/D_1 + (x_j - x_o) D_1/D_2]}{(D_1 D_2)^2} \tag{3.13}$$

$$\frac{\partial g_{ij}}{\partial y_o} = \frac{D_1 D_2 (2y_o - y_i - y_j) + N[(y_i - y_o) D_2/D_1 + (y_j - y_o) D_1/D_2]}{(D_1 D_2)^2} \tag{3.14}$$

$$\frac{\partial g_{ij}}{\partial f} = \frac{D_1 D_2 (2f) - Nf[D_2/D_1 + D_1/D_2]}{(D_1 D_2)^2} \tag{3.15}$$

3.2.3 The Calibration Algorithm Results

The flow chart of the MATLAB program used to solve the problem of calibrating the focal length and the image coordinates offsets is given in figure 3.1.

In order to run the calibration program, the image processing algorithm is applied for the TAURUS CCD image 3.2 to get the star coordinates (x_i, y_i). Manual star identification is done to each one of the centroided stars to have the inertial position vector of all the imaged stars for the initial image.

Figure 3.3 shows the Calibration errors for the bore-sight position (x_o, y_o) and the focal length (f) versus the number of iterations for the TAURUS image. It can be shown that the values of (x_o, y_o) and f are converged after 2 or 3 iterations. The test results showed that the Calibrated Focal length = 64.2964, the Calibrated

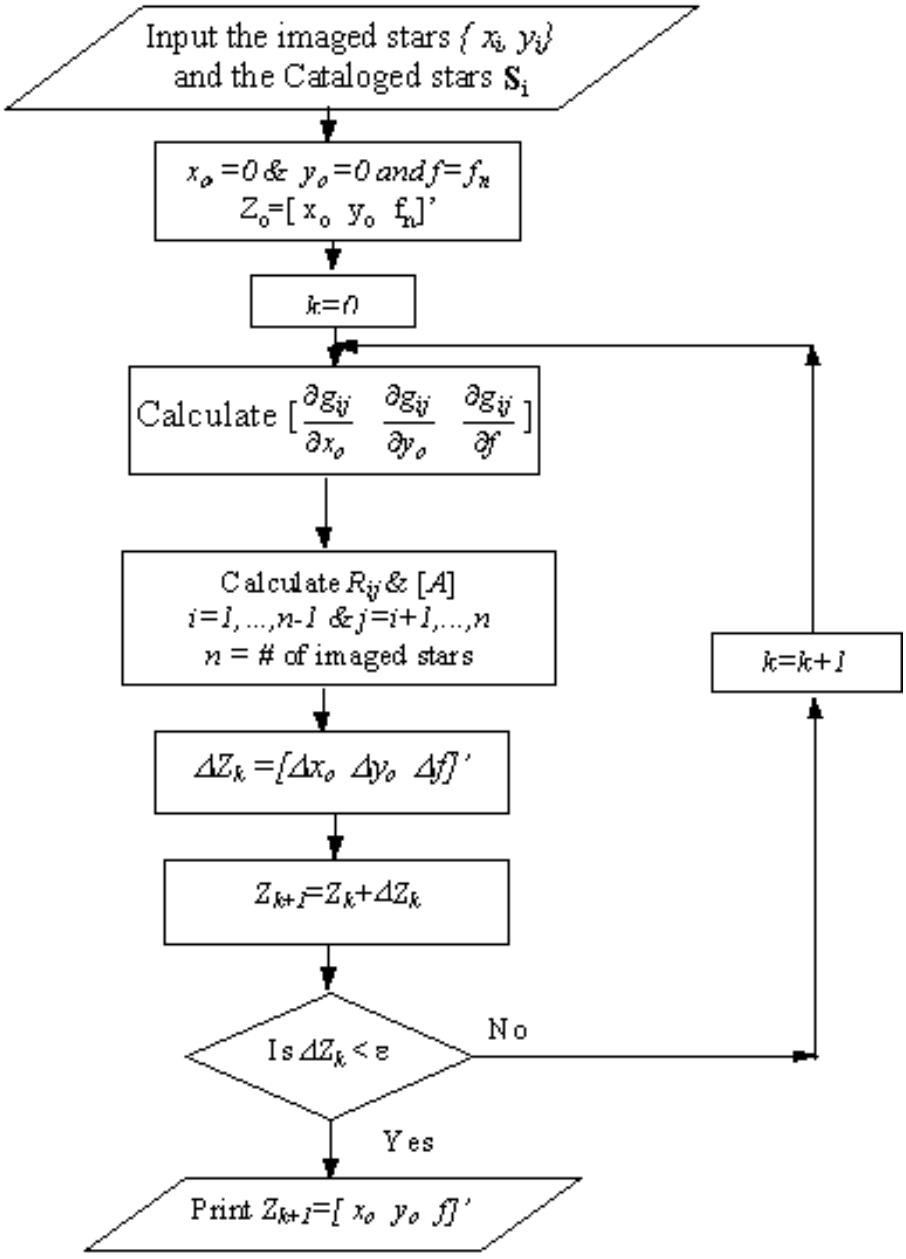

Figure 3.1 Flow chart for the calibration algorithm

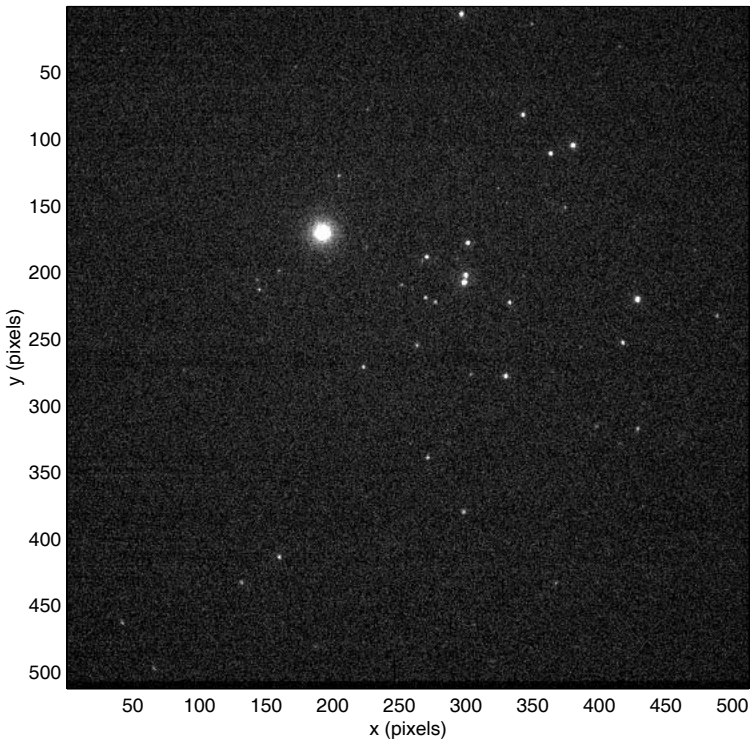

Figure 3.2 Night sky image for TAURUS

Y-axis offset x_o = -0.71896 and the Calibrated X-axis offset y_o = -0.50285. The time consumed by this calibration algorithm was 0.701 sec (using PII 400 Mhz PC).

Figure 3.4 shows the effect of the calibration parameters on the identified stars of the TAURUS image. All the stars are easily and autonomously identified after including the results of the offsets and the focal length on the imaged vectors.

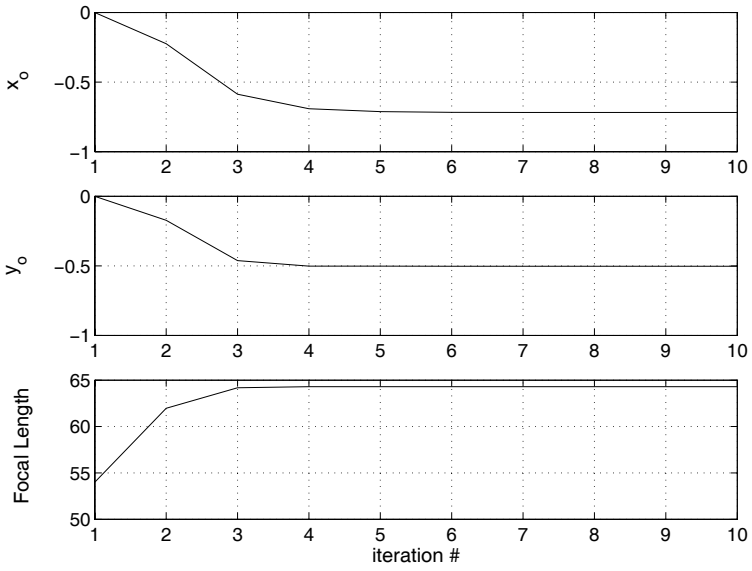

Figure 3.3 The estimation of the offsets and the focal length

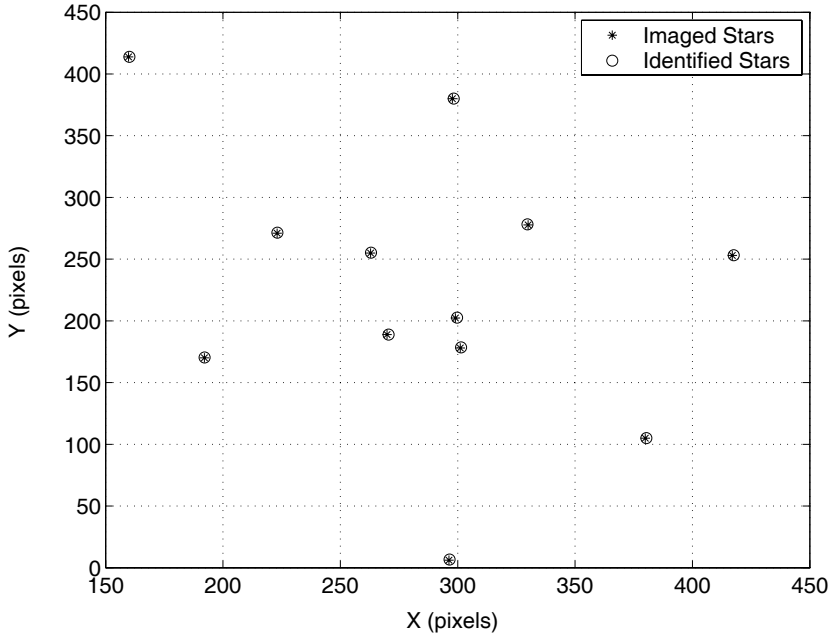

Figure 3.4 The measured and the identified stars for TAURUS

CHAPTER IV

LOST-IN-SPACE PYRAMID ALGORITHM FOR ROBUST STAR PATTERN RECOGNITION

4.1 Introduction

An in-space star-pattern identification capability is becoming an increasingly important aspect of spacecraft navigation. The ability to recognize stars autonomously and to determine spacecraft attitude greatly enhances the value of star-camera data and has many advantages. Spacecraft designed with this inherent autonomy are less reliant on expensive and fragile ground communication links, are more robust against system failure, require fewer sensors and have higher pointing-accuracy capabilities. A robust method is introduced for autonomous star pattern identification (the contents of this chapter are summarized from [18]). The method is demonstrated to be highly efficient and especially, a provably reliable means for solving the general lost in space problem where no prior estimate of pointing is available. At the heart of the method is the k-vector approach for accessing the star catalog, which provides a *searchless* means to obtain all possible cataloged stars from the whole sky that could possibly correspond to a particular measured pair, given the measured interstar angle and the measurement precision. A tiered logical structure, making use of k-vector accessed candidate stars for each measured pair, is introduced where interstar angles for triples and quadruples, and so on star patterns are matched, with an analytical expression derived for the expected frequency of randomly matching these patterns. This expected frequency can be used to rigorously terminate the star pattern matching process with high confidence in the star identification results. In addition, we

introduce a novel way to include in the star identification process binary stars that are too close together to be centroided as distinct stars. All of these developments are supported by simulations and by a few ground test experimental results.

The *Pyramid* algorithm was presented for the first time in the 24th Annual AAS Guidance and Control Conference, January 31 - February 4, 2001, Breckenridge, Colorado [18]. It encountered a immediate appreciation from users, and repeatedly successful on-orbit applications in the High Energy Transient Explorer (HETE) satellite which has been operating in the flight system since July 2002 [27]. Also, Draper Laboratory is currently developing the Inertial Stellar Compass,,a low power stellar inertial attitude determination system [28], which uses the *Pyramid* as the main program for star identification. Also of significance, the method has been adopted for the StarNav dual field of view autonomous star tracker which is the science attitude determination system for the NASA EO-3 (GIFTS) New Millennium mission.

4.2 Star Identification Problem

A high percentage of spurious images (spikes) introduces a crisis in almost all existing algorithms for star pattern recognition for stars imaged by CCD star trackers. Failures and anomalies associated with such spurious images have been experienced in space missions which used star trackers to estimate the spacecraft attitude. For example, the STS 101 SOAR star tracker experiment encountered sun reflections from an adjacent experiments debris, causing a large number of spikes that, in turn, caused the star pattern recognition algorithms (used for SOAR) to fail.

This chapter presents a new star pattern recognition algorithm which is based on a "Pyramid" structure of stars. The "Pyramid" solution, better than any known approach to solve the problem, presents the simultaneous advantages of being ex-

tremely efficient and robust to random spurious images. In fact, the capability to identify spikes (due to electronic noise, planets, light reflections, etc.), is such that the proposed method has been demonstrated to reliably accomplish the star identification process with as few as 4 valid star images and up to 24 random spikes! Of course, this extreme number of spurious images would be a most rare occurrence in practice, but we believe any existing algorithm would encounter reliability difficulties with fewer spikes. Actually, the method presented herein can tolerate even more noise so long as there are at least four valid stars (for the case of modern star trackers with equivalent angle star centroiding errors of a few arc seconds), however, the computation time is obviously a function of the number of spurious images.

In lieu of writing the details of the algorithm, we first summarize the major logical steps and the new features associated with this algorithm. Subsequently, we go into selective detail. The "Pyramid" LISA Algorithm contains several important new features. They are:

(1) Access to the star catalog using the k-vector approach [29], [30], [23], instead of the slower binary search technique. The k-vector database is built a priori for some given working magnitude threshold and for the star tracker maximum angular aperture. Essentially, the k-vector table is a structural database of all cataloged star pairs that could possibly fit in the camera field of view, over the whole sky. The star pairs are ordered with increasing interstar angle. The data stored is the k index, cosine of the interstar angle, the master catalog indices $I[k]$ and $J[k]$ of the k^{th} star pair. The k-vector access logic is invoked in real time for a minimal set of star pairs in elementary measured star polygons (3 for a triangle, 5 for a 4-star pyramid, etc.); the fact that the vertices between adjacent measured star pairs share a common cataloged star is the key observa-

tion leading to logic for efficiently identifying the stars by simply comparing the k-vector accessed catalog indices from the several sets of candidate star pairs (which must contain the common measured pivot star, if it is in the catalog).

(2) Avoidance of identical (redundant) information requests. The information provided by the k-vector for an observed star pair (say, s_i-s_j) are stored so that any further request of such information does not require another identical use of the catalog access. Mainly the stored information are the number of the admissible stars pairs together with the identifiers of the involved catalog stars, information contained in the two integer vectors of indices I and J.

(3) "Smart" choice of sequentially considered measured star triangles. Existing algorithms, which are based on star triangles and which are designed to identify and discard spikes, these algorithms should be extended should consider the non negligible possibility: While scanning all the possible observed star triads by means of three *for* loops, in the case that the star associated with the most external loop is a spike, then most of the consumed time spent is useless. To avoid such unpleasant wasted computation, the combination sequence for the considered triads are devised so that the subsequent triangle choice maximizes the changes in the three indices identifying the triads (rather than the traditional schemes that in essence pivot exhaustively about a possibly invalid star). This is accomplished in the algorithm presented here by pre-storing all of the combinations of n objects taken 3 by 3, with n varying from 3 to 28, arranged in a sequence which maximizes the indices changes on successive triples, and no pivot star is retained for more than two successive instances of matching logic.

(4) Utilization of a robust four star basis "Pyramid", instead of the more classic triangle, to increase the probability of a correct star identification process. This

allows also an easy way to identify spikes. The only limitation consists of the fact that at least four good stars are needed to build the Pyramid. At the precision limit of a few arc seconds achieved by state of the art star trackers, a star identification process that matches measured and cataloged interstar angles for a four star pyramid is essentially a certain match.

(5) Finally, and perhaps most importantly, we introduce an analytical means to compute the "expected frequency of random occurrence" that a cataloged polygon of stars could possibly match, to within camera precision, the given measured polygon. This analytical means of computing the expected frequency is novel and important to eliminate the need for expensive and slowly converging Monte Carlo estimates of star identification reliability.

4.2.1 A Smart Technique to Scan Triangles

The *Pyramid* algorithm is built starting with a basis star triangle. Now, which kind of different approaches are available to scan subsequent triangles in order to find out the first one, on which the *Pyramid* logic can be built on? There is an optimal choice on the subsequent triangle choices. The problem we would like to avoid is to persist using some star, that may be a spike, and not a real star! The right choice of the triangle sequence implies the need to maximize the changes on the index stars of one selection with respect the next one. This maximization defines the optimal sequence. The original *Pyramid* version proposed a heuristic approach which indicated that the sequence obtained by a random shuffling of all the triad combinations, even it will never guaranteed to be optimal, it will statistically avoid retaining spurious stars and wasting the trials to match angles to other stars. To this end, the original *Pyramid* version reads out a file containing the indices of the shuffled sequence triads for

triangle selection. This choice, however, which has been found much more suitable than the crude and simplest approach of three inner loops, presents the disadvantage of requiring additional memory (especially when a high value of the observed stars n is adopted), since all the triangle index combinations, must be memorized. To avoid this problem, the new version of *Pyramid* adopts a smart technique to produce the indices of subsequent star triangles, which is built on the simplest three inner loops concept. This technique, whose results should be compared with respect to the mathematically rigorous optimal solution to this problem (still unknown), is described below using a pseudo-code language, easy to be translated into any another existing programming language.

```
LOOP dj from 1 to (n-2),
  LOOP dk from 1 to (n-dj-1),
    LOOP i from 1 to (n-dj-dk),
      next combination is "[i i+dj i+dj+dk]",
    END LOOP i,
  END LOOP dk,
END LOOP dj,
```

For instance, for $n = 6$ observed stars, the smart sequence of triad indices is given in Table 4.1. It is easy to see that there is no persistence on a given index (star) more than two subsequent times.

4.3 Lost in Space Pyramid Algorithm

The proposed Pyramid LISA algorithm does not use any information on the star magnitude.

Table 4.1 Smart sequence of triad indices for $n = 6$ observed stars

i	1	2	3	4	1	2	3	1	2	1	1	2	3	1	2	1	1	2	1	1
j	2	3	4	5	2	3	4	2	3	2	3	4	5	3	4	3	4	5	4	5
k	3	4	5	6	4	5	6	5	6	6	4	5	6	5	6	6	5	6	6	6

Figure 4.1 shows the basic stars structure used within the Pyramid LISA algorithm, which consists of a basic star triangle, identified by the indices $i,\ j,\ k$, together with a "confirming fourth star", identified by the index r.

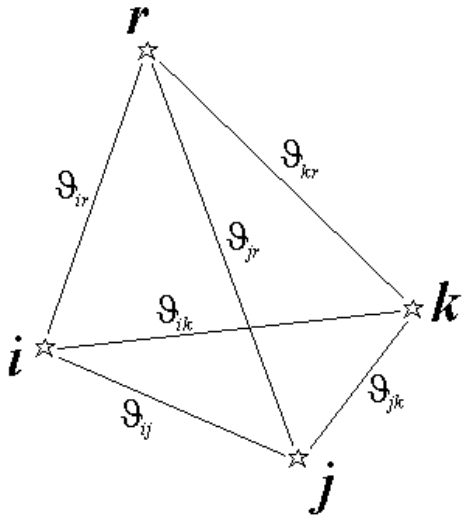

Figure 4.1 Basic star triangle and pyramid

The method, whose flow-chart is given in Fig. 4.2, essentially accomplishes the task by the following steps (where n is the number of observed stars):

(1) if $n = 3$, then the four star Pyramid $[i,\ j,\ k,\ r]$ cannot be built. Therefore the Pyramid logic simply seeks to establish if the triangle is unique. Using formulas derived below, a frequency with which this measured triangle match could be made with a random invalid cataloged triangle is computed; if this number is greater than some tolerance, the star identification will be rejected. Also, if more than one cataloged triangle is found to match the measured triangle to within the measurement tolerance, then the star identification is not accepted as unique.

(2) If $n > 3$, then Pyramid LISA algorithm looks for a unique triangle $[i,\ j,\ k]$ by scanning the "smart" combinations indices associated with all the star triangles, and checking the k-vector accessed indices to establish a hypothesis for the cataloged indices for each star. Also, using formulas derived below, a frequency with which this measured polygon match could be made with a random invalid polygon from the catalog is computed; if this number is greater than some tolerance, the star identification is rejected.

(3) If a high confidence triangle identification in point 2) is found, then Pyramid LISA algorithm will scan the remaining stars to find one which further confirms the basic star triangle $[i,\ j,\ k]$, with the analytical frequency test employed at each stage.

(4) When point 3) is accomplished with success, then the Pyramid is found as that having the star indices $[i,\ j,\ k,\ r]$. This means that these four stars are, at this point, identified with a very high confidence, computable using formulas

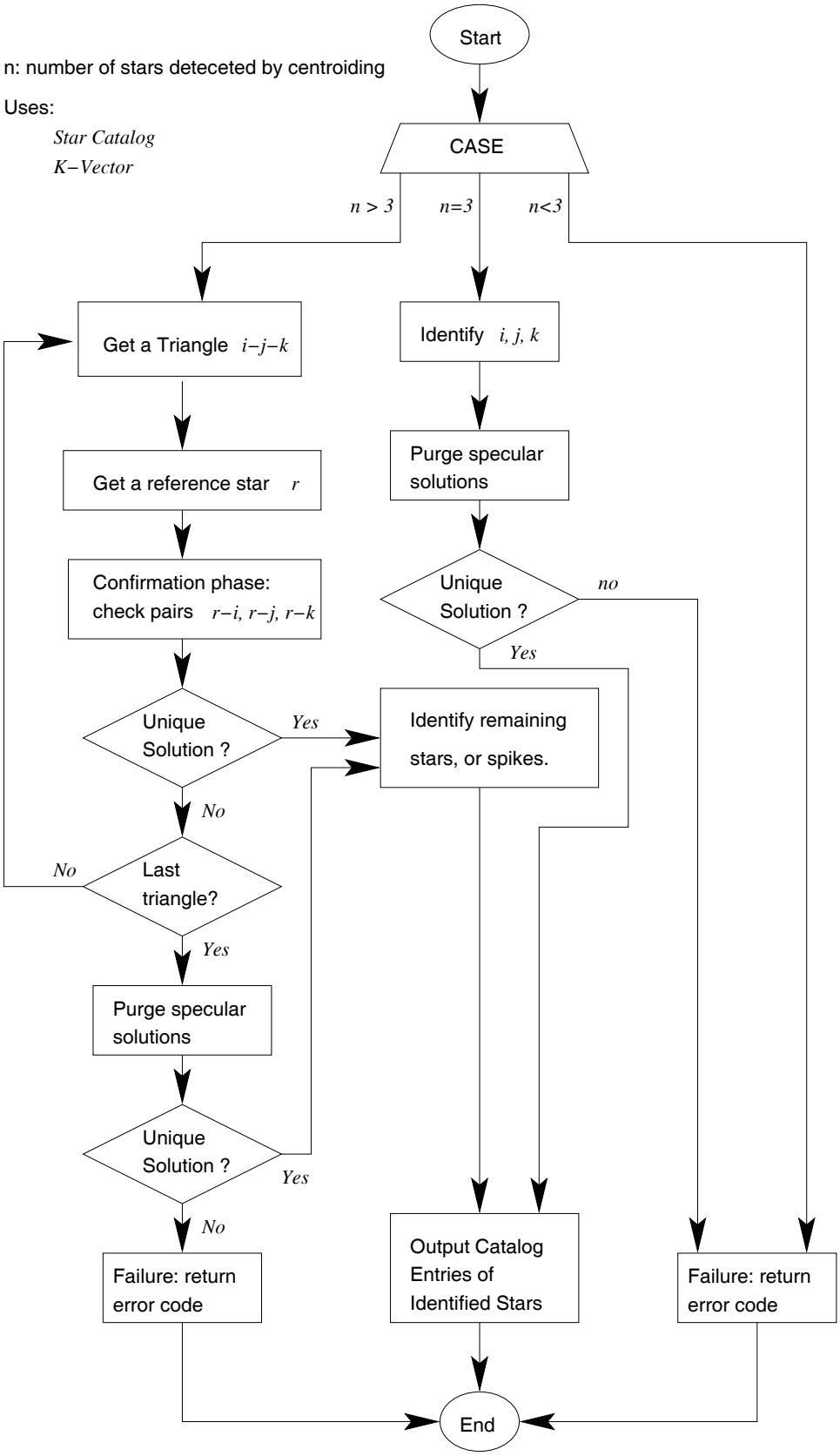

Figure 4.2 Pyramid flow chart

presented below. The three stars constituting the basic star triangle $[i,\ j,\ k]$ are then used to identify the remaining stars (p) as good ones (when the stars confirms the basic star triangle) or to identify the measured image as a spike (otherwise). If desired, the entire set of identified stars could be used to form an n-star polygon and a final frequency can be analytically computed to indicate the likelihood that a random match could match all of the angles to within measurement precision. Typical random frequencies for modern star trackers with four or more valid stars is smaller than 10^{-7}, so matching more four or more stars usually results in near certain star identification, especially if this occurs on successive star identifications and the identified stars have overlap.

(5) If the confirming r star is not found, then another basic star triangle $[i,\ j,\ k]$ is selected by choosing another "smart" combination of star indices. This means to go to step 2).

(6) If all the "smart" combinations of star indices are used, then Pyramid LISA algorithm will provide the basic star triangle $[i,\ j,\ k]$, if unique. Otherwise, Pyramid logic will output a flag indicating a failure in the star identification process. Note our basic philosophy, we establish a level of confidence a priori, and we prefer to report a star identification failure (perhaps once in 1000 images with four or more valid stars), rather than output a lower confidence star ID. Modern attitude estimation algorithms can very easily tolerate infrequent data dropouts, but are generally much less forgiving of invalid star identification. Following a process initiated in chapter 8, we can associate a frequency of mismatch (for a given accuracy level, field of view, and for each number of stars in the pattern). For a 4 star pattern, the probability of mismatch is about 10^{-11}.

The c-code for the *pyramid* algorithm is given in details in appendix A.

Another practical problem which also appears during the star identification process, is generated by the presence of double stars. Most of the catalog double stars have interstar angles smaller than the centroiding spot of light.

This means that the centroiding process identifies the composite image as only one star instead of two. The magnitude information of the observation of such double stars will result as one apparent "star" with a brightness equal to the sum of the star instrument magnitudes.

The star identification process is typically accomplished by trying to match measured and cataloged star pattern figures, as for instance triangles or pyramids. Now, when one of the stars belonging to the selected pattern figure is a double star, then more than one solution may become available, or, more likely, no solution may be possible.

At this point the possibilities are two: 1) to discard the observed double star, or 2) to merge such double stars in the catalog into an equivalent one. How to accomplish this second solution is treated in the next section.

4.4 Double Star Elimination

Double stars create several difficulties, even for the smartest star identification algorithm. The presence of a double star could, in fact, be detected. But what to do at this point? Most of the existing algorithm just identify them and discard them as spikes. If many redundant distinct stars are imaged, then this is good approach, but this may delete crucial information, especially in sparse regions of the sky. In the following we consider replacing the double star in the catalog by an "equivalent" or "composite" star.

To do this, the equivalent star must be defined in both magnitude and direction,

and it is evident that the process used to create the equivalent stars in the catalog and the image processing of the particular star tracker's imaged double stars must be highly consistent.

4.4.1 Magnitude

The magnitude m of a star is ideally related to the brightness (or flux density) F by means of

$$m = m_0 - 2.5 \log F \tag{4.1}$$

that allows us to write

$$F = e^{(m_0 - m)/2.5} \tag{4.2}$$

where m_0 is a scale constant. For the sake of simplicity in the present discussion, we assume that m is the *instrument* magnitude, and we assume that all of the star energy flux, which is a function of wavelength, is measured by the camera. We mention that a more elaborate development can be done to generalize the developments in this section whereby the inputs would be an energy distribution for each star and a transfer function characterizing the responsivity of the particular camera detector array. Convolution of the sensor energy flux with the responsivity of the camera, and integration over time would yield a more formal definition of instrument magnitude. While this more general development would have merit, we leave these developments for future studies and utilize the definitions of Eqs. (4.1, 4.2), to establish the essential idea, and we mention in advance that these elementary developments have been found to have practical utility. One of the issues that encourages approximation in magnitude calculations is the well-known truth that it is difficult for *any* simulation to match a single star camera measurement's instrument magnitude to closer than 10%, thus the primary role of simulation of magnitude to estimate catalog length,

rough ordering of real time logic that is fairly insensitive to the ordering, and so on.

From the definition of Eq. (4.2), two stars with magnitude m_1 and m_2 have an energy flux ratio of

$$\frac{F_1}{F_2} = e^{(m_2 - m_1)/2.5} \tag{4.3}$$

Now, an "equivalent star" for a given double star will have a brightness as the total energy flux of the double stars, that is,

$$F = F_1 + F_2 \tag{4.4}$$

and a corresponding magnitude m that can be evaluated using Eq. (4.3) written for the "equivalent" star and one of the double stars

$$\frac{F}{F_2} = \frac{F_1 + F_2}{F_2} = e^{(m_2 - m)/2.5} \tag{4.5}$$

this equation allows us to evaluate the equivalent magnitude for the equivalent star to replace a double star as

$$m = m_2 - 2.5 \log\left(\frac{F_1 + F_2}{F_2}\right) = m_2 - 2.5 \log\left(1 + \frac{F_1}{F_2}\right) \tag{4.6}$$

and, using Eq. (4.3) we obtain the equivalent magnitude solution in terms of m_1 and m_2

$$m = m_2 - 2.5 \log\left(1 + e^{(m_2 - m_1)/2.5}\right) \tag{4.7}$$

4.4.2 Direction

The minimum angular distance such that stars can be identified as distinct, is associated with the size of the centroiding submatrix or mask. We mention that the mask size is a function of "everything", including star magnitude, integration time, CCD responsivity and so on, and setting the optimal mask size typically requires some experience with the particular camera.

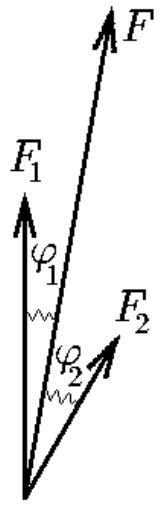

Figure 4.3 The "equivalent" star

Let 3σ characterize the accuracy of a centroided star direction detected by a CCD, determined from analysis of and experience with the particular camera. With this known precision, then the minimum angular distance between two close stars such that their separation can always by observed is adopted as $\beta = 4(3\sigma)$.

For a star tracker having a square FOV ($\vartheta \times \vartheta$), which identifies the defocused stars by an $n_p \times n_p$ sub-matrix and which has an $N \times N$ square CCD, then a star light spot will cover an angular separation just a bit smaller than $\vartheta n_p/N$ per side. Usually, two main approaches are used to select the value for n_p: 1) a fixed value [usually a $n_p = 3$, or $n_p = 5$], or 2) a value of n_p which is a function of the magnitude of the star or, better, which is a function of its instrument brightness.

For instance, with a tracker having $\vartheta = 8$ deg., $3\sigma = 10$ arcsec, $n_p = 5$, and $N = 512$, two stars are observed as distinct if they are displaced by an angle of $\beta = 4(3\sigma) = 40$ arcsec, while two centroiding masks are separated by an angle of $\vartheta n_p/N = 281.25$ arcsec, a value which is seven times β. For these reasons, it is very reasonable to substitute, in the star catalog, all the double stars v_i and v_j such that

$v_i^{\mathrm{T}} v_j > \cos\beta$. We mention that typical star trackers and centroiding algorithms result in a σ corresponding to an equivalent angle subtended by 1/10 to 1/20 of a pixel in the focal plane.

Now the direction of the "equivalent" star v is defined, analogously, as the direction associated with the static equilibrium of two springs having stiffness constant F_i and F_j and displaced one to another by the angle φ, where $\cos\varphi = v_i^{\mathrm{T}} v_j$. In fact the idea is that the "center of brightness" of the double stars is decided by brightness of each star, which is the only parameter which influence the evaluation for the centroiding. Referring to Fig. 4.3, the equilibrium equation is

$$F_1\ \varphi_1 = F_2\ \varphi_2 \tag{4.8}$$

and we can write that

$$\varphi = \varphi_1 + \varphi_2 = \varphi_1\ \left(1 + \frac{F_1}{F_2}\right) = \varphi_1\ \left[1 + e^{(m_2 - m_1)/2.5}\right] \tag{4.9}$$

which allows the evaluation of φ_1, and $\varphi_2 = \varphi - \varphi_1$. Now the direction of the "equivalent" star, since φ, φ_1, and φ_2 are all very small, is

$$v \sin\varphi = v_1 \sin\varphi_2 + v_2 \sin\varphi_1 \cong v_1\varphi_2 + v_2\varphi_1 \tag{4.10}$$

which completes the solution. Results of this process, for all stars with magnitude less than 6.3, is given in Table 4.2. We mention, that while the above handling of double stars looks promising, we are continuing to work this issue. Certainly, if there are adequate redundant stars imaged and cataloged, one can simply delete from the catalog all pairs closer than a threshold, and these will be automatically deleted from the measured stars because they will almost certainly not be identified by the star identification algorithm (i.e., they will appear to be spurious spikes in the image).

Table 4.2 Equivalent double star centroids

m_i	m_j	m	φ	φ_i	φ_j		m_i	m_j	m	φ	φ_i	φ_j
1.4	3.8	1.29	4.1593	0.4152	3.7441		4.4	6.4	4.24	20.1658	2.7815	17.3843
1.4	5.4	1.37	5.0177	0.1253	4.8924		4.6	5.8	4.29	3.7859	0.9458	2.8401
1.5	5.1	1.46	9.5025	0.3385	9.1640		4.7	6.0	4.41	23.1232	5.3888	17.7344
2.0	4.0	1.84	15.1999	2.0965	13.1033		5.0	6.0	4.63	2.4725	0.7064	1.7661
2.4	5.1	2.31	33.9352	2.6368	31.2984		5.1	5.4	4.48	18.7387	8.0896	10.6491
2.9	5.7	2.82	19.3805	1.3835	17.9970		5.1	5.4	4.48	6.6071	2.8523	3.7548
3.1	5.9	3.02	20.5505	1.4670	19.0835		5.1	6.3	4.79	2.7441	0.6855	2.0585
3.5	6.1	3.40	29.9239	2.5294	27.3945		5.1	6.0	4.70	6.5023	1.9817	4.5206
3.6	5.2	3.37	4.0976	0.7685	3.3291		5.2	5.8	4.70	21.4823	7.8607	13.6216
3.8	5.2	3.53	10.9892	2.3855	8.6038		5.4	5.6	4.74	29.4056	13.3594	16.0463
3.8	4.7	3.40	9.2002	2.8040	6.3962		5.5	6.4	5.10	10.6997	3.2610	7.4387
3.9	4.8	3.50	7.8762	2.4005	5.4757		5.5	5.8	4.88	39.5459	17.0723	22.4736
3.9	5.3	3.63	8.7954	1.9092	6.8862		5.5	6.3	5.07	11.1819	3.6288	7.5530
4.1	5.2	3.76	4.9973	1.3362	3.6611		5.8	5.8	5.04	14.1344	7.0672	7.0672
4.2	5.6	3.93	27.0689	5.8759	21.1930		5.8	5.9	5.09	0.7339	0.3502	0.3838
4.2	4.9	3.74	23.2722	8.0273	15.2449		5.9	6.4	5.37	21.8031	8.4471	13.3560
4.2	5.5	3.91	4.5777	1.0668	3.5109		5.9	5.9	5.14	22.1001	11.0500	11.0500
4.2	6.1	4.02	3.7345	0.5572	3.1773		6.0	6.4	5.43	14.5550	5.9586	8.5964
4.2	4.6	3.63	4.1295	1.6905	2.4389		6.0	6.4	5.43	2.9184	1.1947	1.7236
4.3	5.6	4.01	5.9160	1.3787	4.5372		6.0	6.3	5.38	38.1098	16.4523	21.6575
4.3	5.5	3.99	30.1130	7.5230	22.5901		6.1	6.3	5.44	3.3101	1.5038	1.8063
4.3	6.3	4.14	6.9888	0.9639	6.0248		6.2	6.3	5.49	3.3646	1.6053	1.7593
4.4	5.2	3.97	15.3792	4.9910	10.3882		6.2	6.3	5.49	20.8926	9.9680	10.9246

4.5 *Pyramid* Algorithm Results

The *Pyramid* LISA algorithm, introduced in this paper, is programmed using many computer languages like MATLAB, C/C++ and JAVA. It has proven very successful for night sky tests and also for monte carlo star image simulations.

4.5.1 Using Monte-Carlo Simulations

End to end numerical tests, based on simulations of random unknown spacecraft attitude, star image centroid measurements (including Gaussian measurement errors with zero mean and 17μ rad standard deviation), star catalog access, Star-ID, and attitude estimation have been carried out. We add a very high probability of having spikes in each simulated star image. Spikes can appear in the real images due to electronic noise, planets, light reflections, etc. In fact, the *Pyramid* algorithm has been demonstrated to reliably accomplish the star identification process with as few as 4 valid star images and up to 24 random spikes. Of course, this extreme number of spurious images would be a most rare occurrence in practice, but we believe any existing algorithm would encounter reliability difficulties with fewer spikes. Actually, all the simulated images can be tolerated even with more noise so long as there are at least four valid stars (for the case of modern star trackers with equivalent angle star centroiding errors of a few arc seconds), however, the computation time is obviously a function of the number of spurious images.

A 1000 star image set of observation data were simulated using random attitude and the true known star information from the catalog, together with random numbers to simulate centroiding errors of 17μ-radians (1-σ). Figure 4.4 shows the histogram of the number of star occurrence for the simulated star measurements along with the number of spikes in each image. The execution time for Star-ID for each star image,

using a MATLAB program operated on a PC 450MHz operating under Windows 98, is also shown in Fig. 4.5. This plot gives a meaningful measure of computational that of course can be reduced by over one order of magnitude by utilizing compiled code from, for example, equivalent C-code and an appropriate real time operating system.

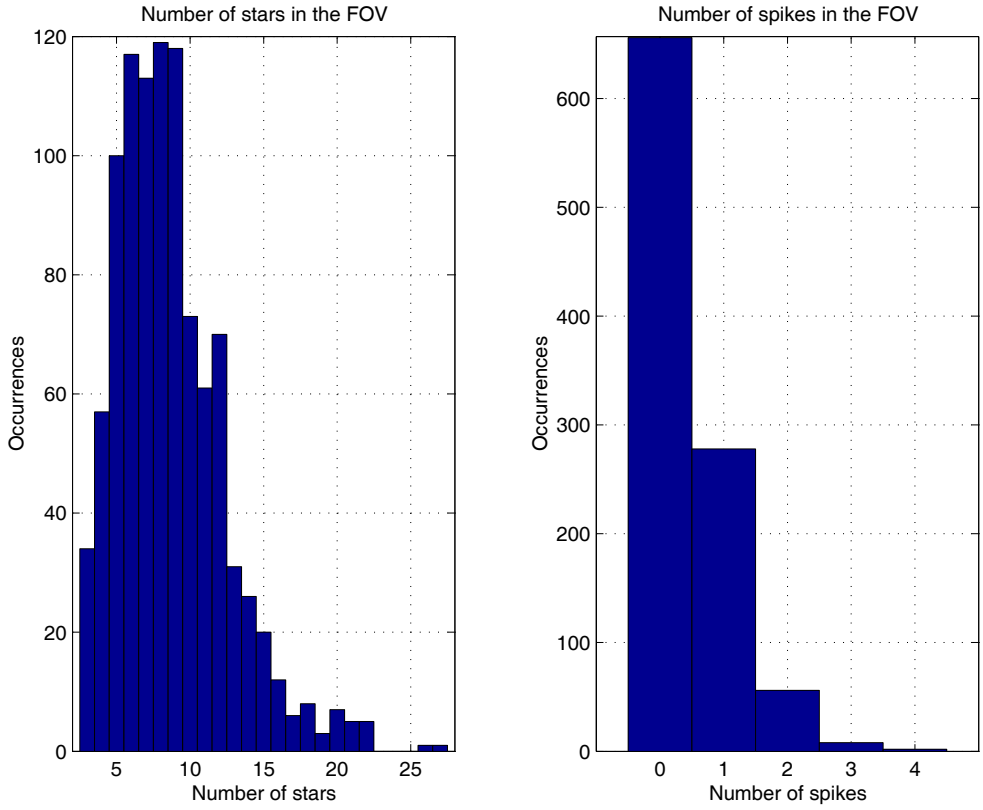

Figure 4.4 Histogram of the number of star and spike occurrences

Now, by using the fastest available attitude estimator (that is, ESOQ-2 [20], with the latest improvement [31]), the estimated attitude direction cosine matrix $C_E(t)$ is calculated using the observed star vectors and the cataloged star vectors of the identified stars [32]. The parameter that quantifies the accuracy of one attitude with respect another one (as, for instance, for the true and the estimated attitudes), is described by the maximum direction error ($\max\{\varepsilon\}$), or the expectation ($E\{\varepsilon\}$) of

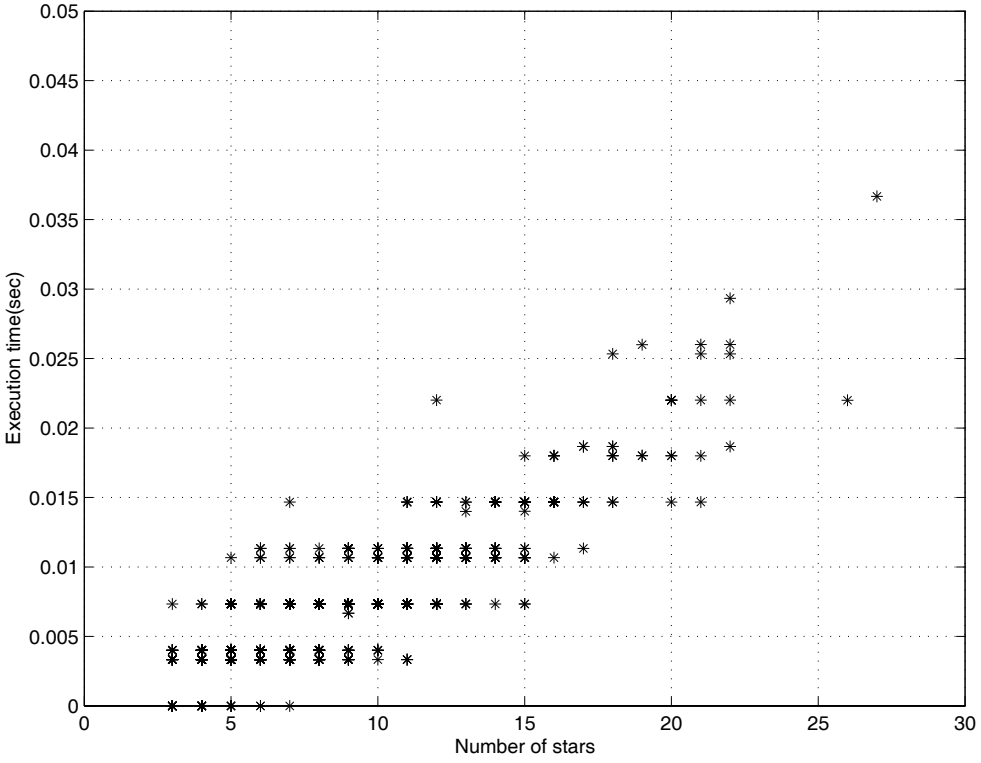

Figure 4.5 The execution time versus the number of stars

the direction error, which are evaluated according to

$$\max\{\varepsilon\} = \cos^{-1}\left(\frac{\text{trace}[C_T\, C_E^{\mathrm{T}}] - 1}{2}\right) \qquad \text{and} \qquad E\{\varepsilon\} = \frac{\pi}{4}\ \max\{\varepsilon\} \tag{4.11}$$

Now, during simulation, the true attitude C_T is known. This allows us to conveniently describe the error of C_E, provided by Eq. (4.11), by three different meaningful components. These errors are: 1) the error ε_{oa} of the Optical Axis (OA), and 2) the error range, $\min(\varepsilon_n)$ and $\max(\varepsilon_n)$, experienced by the directions orthogonal to the OA. These errors describe the polarization of the attitude error about the OA when the attitude is estimated using a single FOV star tracker, and fully justifies why the multiple FOVs star trackers have been proposed. From a mathematical point of view,

these errors are evaluated as

$$\varepsilon_{oa} = \cos^{-1}(\, b_{oa}^{\mathrm{T}} \Delta b_{oa}\,) \qquad \text{and} \qquad \begin{cases} \min(\varepsilon_n) = \cos^{-1}(\, b_n^{\mathrm{T}} \Delta b_n\,) \\ \max(\varepsilon_n) \equiv \max\{\varepsilon\} \end{cases} \tag{4.12}$$

where

$$\Delta = C_T\, C_E^{\mathrm{T}} \qquad \text{and} \qquad b_n = \frac{e - (e^{\mathrm{T}} b_{oa}) b_{oa}}{\sqrt{1 - (e^{\mathrm{T}} b_{oa})^2}} \tag{4.13}$$

In particular, b_{oa} identifies the on-board direction of the OA, Δ indicates the corrective attitude matrix, and e its principal axis. The numerical values of ε_{oa} and $[\min\{\varepsilon_n\},\ \max\{\varepsilon_n\}]$, obtained by numerical tests, are shown in Fig. 4.6, Fig. 4.7 and Fig. 4.8 respectively.

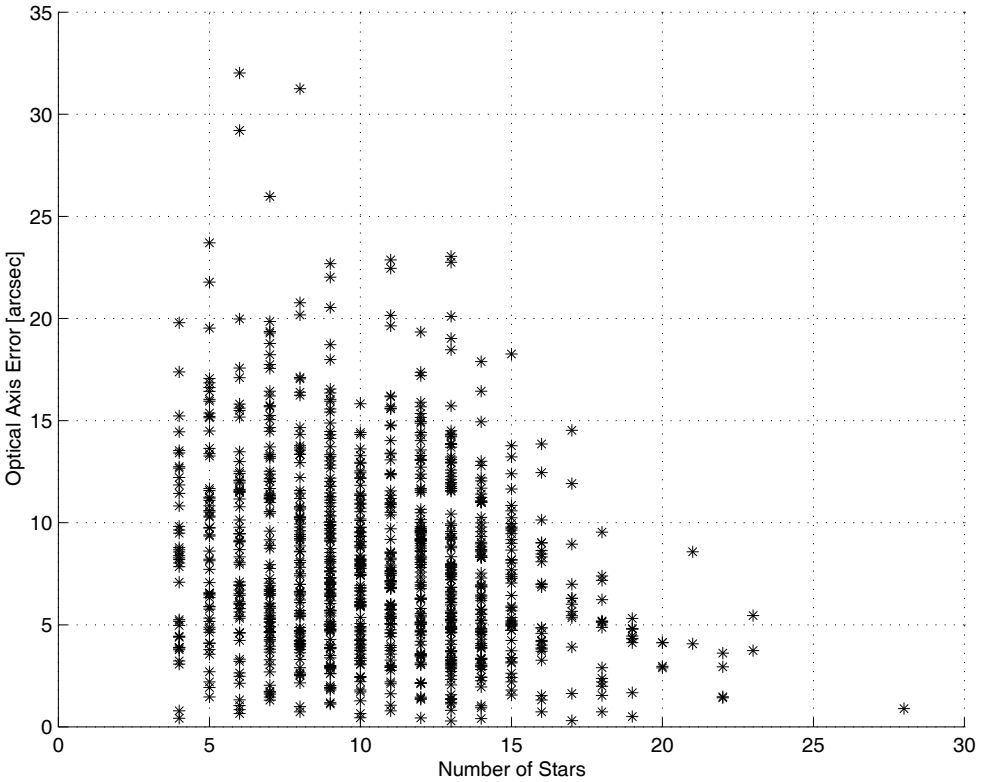

Figure 4.6 Mean errors of the optical axis

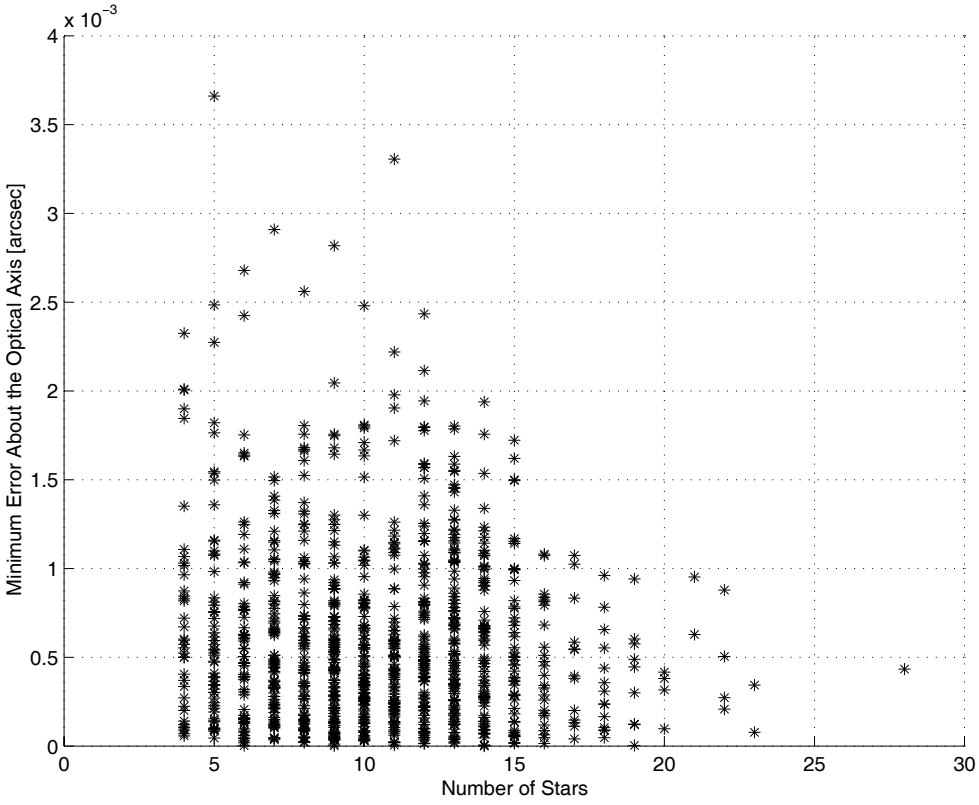

Figure 4.7 Minimum errors about the optical axis

The percentage of successful tests was in the order of 99.5% out of the 1000 tests. These tests made evident that if there are only three stars in the Field of View, there is a higher chance of failure. Besides, with a higher number of stars in the catalog, up to 5.8 Visual Magnitude, the *k-vector* access routine returns a larger number of pairs, and requires a bigger memory structure (from the original $1.6Kb$ to $3.2Kb$, on the average). In order to further reduce the failure rate one can use dynamic memory allocation, but unfortunately not all hardware platforms allows use of this technique.

4.5.2 Using Night-Sky Tests

The *Pyramid* algorithm has been extensively and successfully tested on-orbit [27] and in night sky experiments. For the latter a VC51 Camera (752×582 CCD pixels of

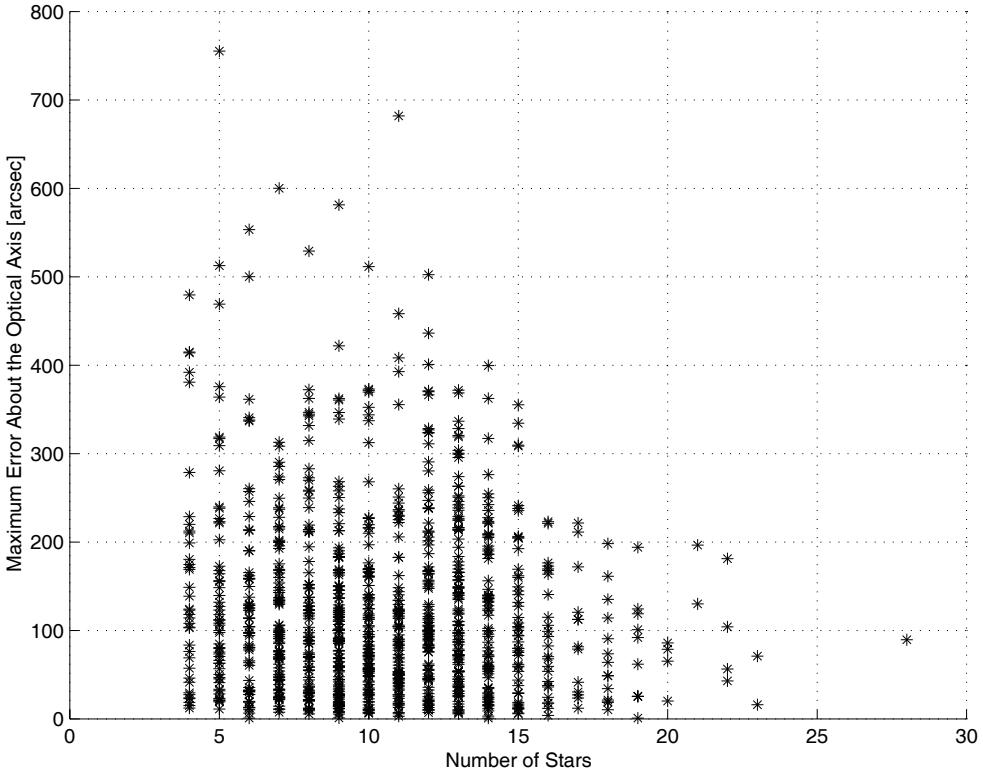

Figure 4.8 Maximum errors about the optical axis

6.5μm×6.25μm size), has been used. The VC51 is equipped with an ADSP2181 32 MHz digital processor, a memory of $16K \times 16$ bits for programs and $16K \times 24$ bits of data. For the image storage, and processing, the DSP is connected with a BUS to a DRAM of 8 MBytes. It is possible to save persistent data on board the camera using a 2 Mbytes EPROM. The lenses used are common Nikon lenses, connected through a C mount adapter. Before performing the tests, the correct focal length f of the lens is evaluated thanks to the new Non-Dimensional Star-ID algorithm [24], capable to identify the observed stars even with a lens whose focal length is completely unknown.

Additional night sky tests of *Pyramid* have been performed with the GIFTS prototype camera "Star1000 1024×1024 pixels". For example, Fig. 4.9 shows a typical image taken by the Star1000 camera. The coordinates of the nine brightest

stars done by the centroiding algorithm [16] as shown in Fig. 4.10. The results of the *Pyramid* star identification algorithm for the Star1000 image given in Fig. 4.9 are shown in table 4.3.

Figure 4.9 Star image using Star1000 camera

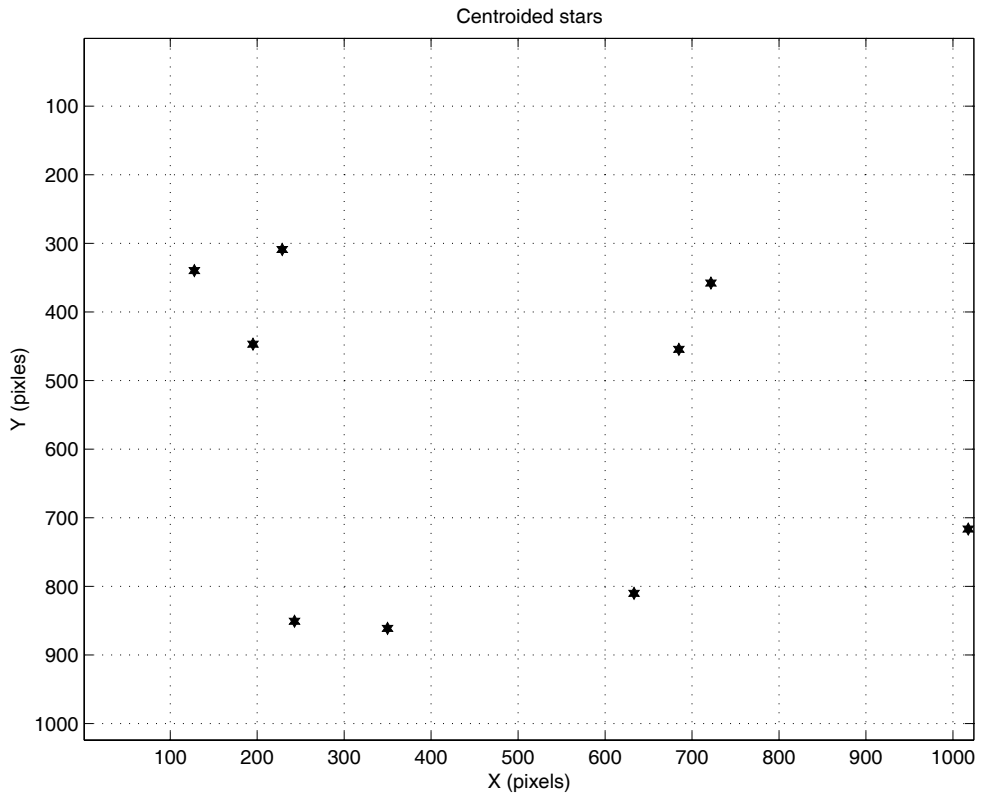

Figure 4.10 The centroiding results for the Star1000 image

Table 4.3 Star identification results for Star1000 image

Star #	Right ascension			Declination			Mn	**ID**
	Hr	M	S	Deg	Min	Sec		
1	2	50	9	27	16	26	3.6	**1251**
2	3	20	31	29	3	37	4.5	**1255**
3	3	18	55	34	14	5	4.8	**945**
4	2	51	42	35	4	24	4.5	**936**
5	2	48	5	29	15	38	4.5	**1250**
6	3	44	30	32	17	55	3.8	**947**
7	2	59	15	35	11	47	4.9	**940**
8	2	43	37	27	43	15	4.7	**1249**
9	3	22	23	27	37	10	5.5	**1257**

CHAPTER V

RECURSIVE MODE STAR IDENTIFICATION ALGORITHMS

5.1 Introduction

Star identification can be accomplished by several different available algorithms that identify the stars observed by a star tracker. However, efficiency and reliability remain key issues and the availability of new active pixel cameras requires new approaches. In this chapter, two novel algorithms for recursive mode star identification are presented (the contents of this chapter are summarized from [19]. The first approach is derived from the Spherical Polygon Search algorithm, here used to access all the cataloged stars observed by the sensor field of view and recursively add/remove candidate cataloged stars according the predicted image motion induced by camera attitude dynamics. Star identification is then accomplished by a star pattern matching technique which identifies the observed stars from the reference catalog. The second method uses star neighborhood information and a cataloged neighborhood pointer matrix to access the star catalog. In the recursive star identification process, and under the assumption of "slow" attitude dynamics, only the stars in the neighborhood of previously identified stars are considered for star identification in the successive frames. Numerical tests were performed to validate the absolute and relative efficiency of the proposed methods. The most critical stage of the attitude determination process using star tracker is the star identification process (Star-ID). This is reflected by a wide variety of different approaches, well described by an existing very rich literature (just to mention some of them, see [18], [33], [34], [35] and [36]. Solving the star-ID problem is traditionally a time consuming process, requiring a large amount of memory

for the star catalog. This is why of faster techniques are still demanded, especially to solve the *Lost-in-Space* problem, where no prior attitude information is available.

The introduction of a new method, the k-vector, to solve the Range Searching Problem without a searching phase (that is, in a way dramatically faster than the standard Binary Search Technique), resulted into a substantial step toward faster algorithms to solve the Star-ID problem. This method, which has been used and demonstrated to effectively access the star catalog data (see [30]), was then specialized in [23] for its application to different problems requiring range searching, yields also to the development of new approaches (see [37,38] and [18]) to solve the general Star-ID problem.

However, the need to solve the *Lost-in-Space* problem, is a generally rare event in the lifetime of a standard operating star tracker. While rarely needed the *Lost-in-Space* solution must be reliable. Most of the time, a star tracker already knows, within an approximate precision, the three axis attitude. This information is, therefore, very useful to alleviate the computations of the star-ID process by reducing the overall star catalog to a small subset of it. Prior knowledge, even approximate, of the angular velocity vector in both magnitude and direction, can also be used to estimate which stars will leave the sensor field-of-view (FOV) and which stars will appear. Using this information, the *recursive* mode is the operating mode which does image processing, star identification, and attitude estimation at a high speed. In fact, the higher attitude knowledge update rate the more effective is this approach.

In this chapter, two new very fast algorithms to accomplish the recursive Star-ID process (that is, not to solve the *Lost-in-Space* problem), are presented. These two methods, which are ideally suited for the emerging active pixel camera technology, are:

- the **Spherical Polygon Search** (SP-Search) approach. This method, which is derived from [37, 38], accesses the stars that can potentially lie within the star tracker. Then the interstar angles between the measured stars and the cataloged stars are used in a tiered logical algorithm to establish the Star-ID, and

- the **Star Neighborhood Approach** (SNA). This method, which represents an alternative to the SP-Search, also accesses candidate stars and performs the Star-ID by locating the observed cataloged stars (those falling in the instrument FOV) by a cataloged knowledge stars neighboring the identified stars from the previous frame.

To start the recursive mode of the star-ID process, the recently developed "Pyramid" Lost-In-Space Algorithm (LISA), presented in [18], which improves the first version of it [39], is adopted to obtain the initial quaternion and the direction cosine matrix. In the "Pyramid" LISA, the Star-ID process is quantified by evaluating the expected random frequencies associated with matching interstar angles from measured star polyhedra.

For each successive frame, the SNA or the SP-Search algorithm, as proposed herein, are used to obtain the expected stars in the FOV, from the star catalog. A "hypothesis-test" logical method for star pattern identification, by matching interstar angles in the measured frame to those in the expected stars, completes the Star-ID process. The angular velocity can be estimated by using the rate gyro and the quaternion is integrated at each time t to obtain the direction cosine matrix, although as frame rates increase, it is possible to dispense with gyro measurements for many missions. The attitude estimator adopted in both the proposed algorithms is the

second "EStimator of the Optimal Quaternion" (ESOQ-2, [20])[1] improved by the modifications described in [31].

The two new recursive Star-ID algorithms, here presented, are supported by simulation results.

5.2 Spherical Polygon Approach

Mortari [37] established the Spherical-Polygon Search approach, to access and identify stars observed by a wide field-of-view (FOV) star tracker. The SP-Search Approach is used to identify the stars as observed by a wide FOV star tracker, uses more than once the k-vector technique [41, 42]. SP-Search does not require any (accurate or not) initial guess of the spacecraft attitude, does not use the typically low accuracy magnitude information. The method uses reference observed star pairs as the basis on which the Star-ID process is accomplished for all of the other and remaining observed stars. Of course matching a single star pair leads to a large number of ambiguities, however, we find matching a four star pattern virtually eliminates the possibility of ambiguities (for > 5000 candidate stars and $\cong 17\mu$ rad. centroiding precision, probabilities of invalid matches are on the order of $\sim 10^{-12}$).

The problem to find all the stars admissible with a given direction $\mathbf{w}_k \equiv \{x,\, y,\, z\}^{\mathrm{T}}$ with uncertainty $h\sigma$, that is, all those falling within the cone of axis $\mathbf{w}_k$ and aperture $h\sigma$, can be easily accomplished using the k-vector technique applied to the three star direction cosine components x, y, and z. In fact, the uncertainty cone of the $\mathbf{w}_k \equiv \{x,\, y,\, z\}^{\mathrm{T}}$ direction implies that the true component $x_{true} = \mathbf{v}_k(1)$ must fall within the range [$s = \sin(h\sigma)$ and $c = \cos(h\sigma)$]

$$x_{true} = [\, cx - s\sqrt{1-x^2},\; cx + s\sqrt{1-x^2}\,] \equiv [x_{min},\; x_{max}] \tag{5.1}$$

[1] ESOQ-2 improves the first (ESOQ-1 or ESOQ), presented in [40].

and, similarly for the other two components y_{true} and z_{true} of $\mathbf{s}_k$, within the ranges

$$\begin{cases} y_{true} = [\, cy - s\sqrt{1-y^2},\ cy + s\sqrt{1-y^2}\,] \equiv [y_{min},\ y_{max}] \\ z_{true} = [\, cz - s\sqrt{1-z^2},\ cz + s\sqrt{1-z^2}\,] \equiv [z_{min},\ z_{max}] \end{cases} \tag{5.2}$$

The stars satisfying the conditions given in Eqs. (5.1, 5.2) are distributed in an annular spherical surface identified as the area between the two cones having axis as the $\mathbf{x}$ coordinate axis and with aperture $\cos^{-1}(x_{min})$ and $\cos^{-1}(x_{max})$, respectively. Now the admissible stars are all those satisfying all the three different and orthogonal annular spherical surfaces (with respect to all of the three coordinate axes).

Thus, the searched admissible stars are all those falling within the intersection among the three annular spherical surfaces depicted in Fig. 5.1.

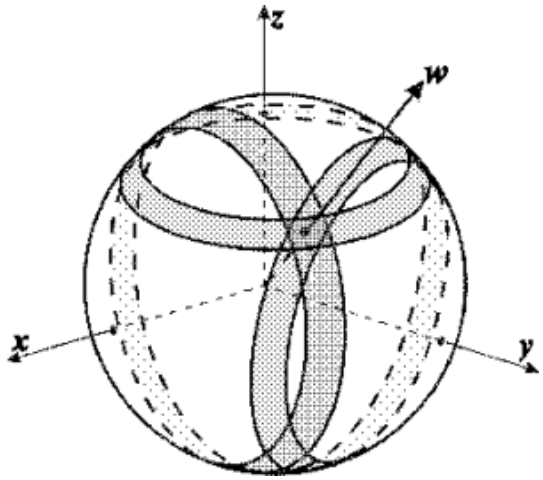

Figure 5.1 The spherical polygon

5.3 Star Neighborhood Approach

The autonomous Star-ID may use the knowledge of the previous frame data to obtain the current frame Star-ID. The SNA is introduced as a novel method of using the previously identified stars to access candidates to match with the current measured

stars. Figure 5.2 shows the representation of the star neighborhood approach in which the union of the neighborhood of previously identified stars is used as the current estimated Star-ID. The radius of the region of the neighbor is depending on the angular velocity of the spacecraft and the frame rate. Pointers to a cone of neighboring cataloged stars are included in the mission star catalog, so once a star has been identified we have a searchless method to access all neighbors in a cone centered on that star.

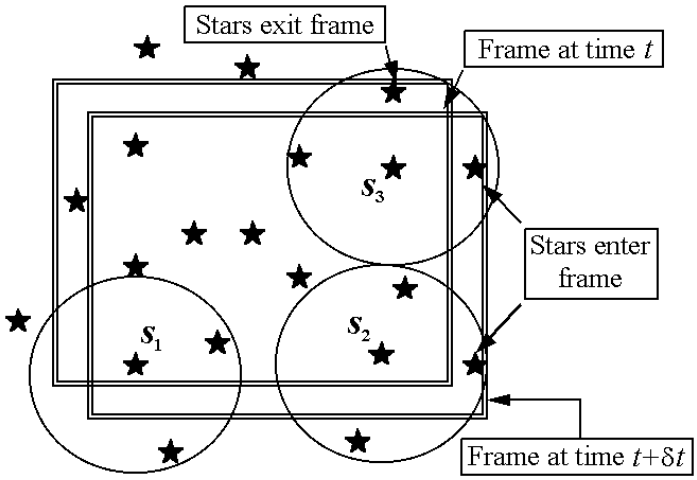

Figure 5.2 Star neighborhood approach

Table 5.1 shows a sample data of some stars that have cataloged indices and the corresponding neighbor indices of (pointers identifying) nine neighboring stars in a cone centered on each one of them.

The star neighbor approach is depending mainly on the locations of the stars with respect to the star tracker sensors four corners in the body reference frame. Figure 5.3 shows the sensor frame in the body-frame.

The sensor frame components of the position vector of the sensors four corners

Table 5.1 The indices of the neighborhood stars

Star #	$\mathbf{v}_1$	$\mathbf{v}_2$	$\mathbf{v}_3$	$\mathbf{v}_4$	$\mathbf{v}_5$	$\mathbf{v}_6$	$\mathbf{v}_7$	$\mathbf{v}_8$	$\mathbf{v}_9$
3341	4132	4579	4580	4131	2145	1257	712	0	0
3342	4161	3707	1555	902	903	2674	43	2146	435
3343	267	553	795	988	1249	1385	1376	1733	1898
3344	208	237	410	1003	1137	1258	1385	1742	0
3345	4634	4140	4141	4590	3711	1120	433	2972	638
3346	54	60	154	171	205	272	617	0	0
3349	4640	4639	1007	802	909	995	1401	1935	0
3350	4640	3715	3746	4147	1386	1735	1918	1559	619
3352	3354	4676	4644	4643	3353	3750	1403	2391	0
3353	62	804	911	996	1403	2152	2392	2391	2700
3354	4644	3352	2393	437	2700	1739	996	807	0
4140	60	154	205	617	893	4141	3346	3345	2972
4141	60	171	205	617	3345	3712	3711	4140	3346

are

$$\begin{cases} \mathbf{s}_1^{\mathrm{T}} = \{ -\sin\vartheta \quad +\cos\vartheta\sin\varepsilon \quad +\cos\vartheta\cos\varepsilon \} \\ \mathbf{s}_2^{\mathrm{T}} = \{ +\sin\vartheta \quad +\cos\vartheta\sin\varepsilon \quad +\cos\vartheta\cos\varepsilon \} \\ \mathbf{s}_3^{\mathrm{T}} = \{ +\sin\vartheta \quad -\cos\vartheta\sin\varepsilon \quad +\cos\vartheta\cos\varepsilon \} \\ \mathbf{s}_4^{\mathrm{T}} = \{ -\sin\vartheta \quad -\cos\vartheta\sin\varepsilon \quad +\cos\vartheta\cos\varepsilon \} \end{cases} \tag{5.3}$$

where $(\pm\vartheta)$ and $(\pm\varepsilon)$ are the right ascension and declination of the sensors four corner. We can use the vector dot product to get the relationships

$$\begin{cases} \mathbf{s}_1^{\mathrm{T}}\mathbf{s}_4 = \mathbf{s}_2^{\mathrm{T}}\mathbf{s}_3 = \cos\vartheta_y = \sin^2\vartheta + \cos^2\vartheta(\cos^2\varepsilon - \sin^2\varepsilon) \\ \mathbf{s}_1^{\mathrm{T}}\mathbf{s}_2 = \mathbf{s}_3^{\mathrm{T}}\mathbf{s}_4 = \cos\vartheta_x = -\sin^2\vartheta + \cos^2\vartheta = 2\cos^2\vartheta - 1 = \cos(2\vartheta) \end{cases} \tag{5.4}$$

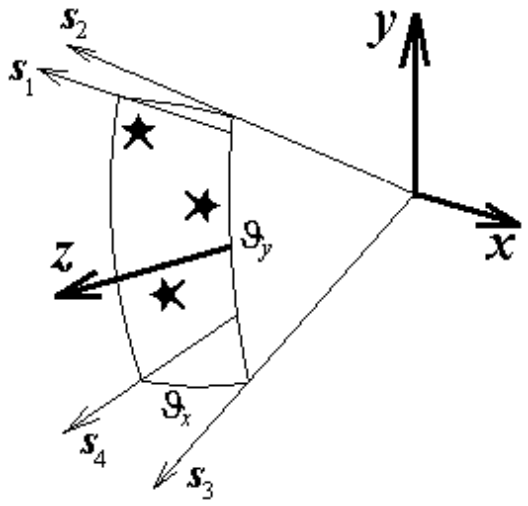

Figure 5.3 The sensor's four corners

where ϑ_x and ϑ_y are the camera angular field of view in $\mathbf{x}$ and $\mathbf{y}$ direction respectively. Thus, we have

$$2\cos^2\vartheta - 1 = \cos\vartheta_x, \quad \vartheta = \frac{\vartheta_x}{2}, \quad \text{and} \quad \cos^2\varepsilon = \frac{\cos\vartheta_x + \cos\vartheta_y}{\cos\vartheta_x + 1} \tag{5.5}$$

For square FOV ($\vartheta_x = \vartheta_y$) we have $\vartheta = \varepsilon$. Now, to check that a certain star is found inside the FOV we have to check the angle between the star position vector $\mathbf{s}_k$ and the normal to each side of the star tracker, i.e. if $\mathbf{n}_{ij}$ is the vector normal to the side of the vectors $\mathbf{s}_i$ and $\mathbf{s}_j$ or $\mathbf{n}_{ij} = (\mathbf{s}_i \times \mathbf{s}_j)/\parallel \mathbf{s}_i \times \mathbf{s}_j \parallel$ then the star $\mathbf{s}_k$ lies inside the FOV if and only if

$$\mathbf{s}_k^{\mathrm{T}}\mathbf{n}_{12} < 0, \quad \mathbf{s}_k^{\mathrm{T}}\mathbf{n}_{23} < 0, \quad \mathbf{s}_k^{\mathrm{T}}\mathbf{n}_{34} < 0, \quad \text{and} \quad \mathbf{s}_k^{\mathrm{T}}\mathbf{n}_{41} < 0 \tag{5.6}$$

So, by knowing the attitude matrix at each time step we can calculate the normal vectors to the sensor frame in the body frame. Meanwhile, using Eq. (5.6) and the star catalog, we can simulate the frames of the stars at each time step.

5.3.1 Star Identification Algorithm

Once the stars are simulated, using the conditions in Eqs. (5.3-5.6), the star neighbor algorithm is used as follows:

1. Using the recent developed "Pyramid LISA" algorithm [18] to solve the Lost-In-Space case. The Star-ID at time to become known.

2. The attitude matrix at each time step is obtained by introducing the angular velocity. The angular velocity is typically obtained by using the rate gyro and the quaternion is integrated at each time t to obtain the direction cosine matrix.

3. Given the attitude matrix $C^{\mathrm{T}} \equiv [\,\mathbf{c}_1 \;\; \mathbf{c}_2 \;\; \mathbf{c}_3\,]$ at time t and the simulated star vectors $\mathbf{s}_i$, $(i = 1, \ldots, n)$. Calculate the star locations $(x_i,\, y_i)$ by using

$$x_i = -f\,\frac{\mathbf{c}_1^{\mathrm{T}}\mathbf{s}_i}{\mathbf{c}_3^{\mathrm{T}}\mathbf{s}_i} \qquad \text{and} \qquad y_i = -f\,\frac{\mathbf{c}_2^{\mathrm{T}}\mathbf{s}_i}{\mathbf{c}_3^{\mathrm{T}}\mathbf{s}_i} \tag{5.7}$$

   where, f is the camera focal length.

4. The estimated location of star i at the current frame, $(x_i,\, y_i)_{t+\delta t}$, is calculated by adding the location of star i at the previous frame, $(x_i,\, y_i)_t$, and estimated translation of the frame $\omega\delta\mathbf{t}$.

5. If the distance between the estimated location of star i $(x_i,\, y_i)_{t+\delta t}$, and the measured location $(x_i,\, y_i)_{measured}$ is less than 3 $(\sqrt{2}\sigma)$ then the star i should have the same ID as in the previous frame. Figure 5.4 shows a representation of stars at time $(t + \delta t)$ and time t.

6. For the unmatched stars from the steps 4 & 5 we use the neighborhood matrix (Table 5.1) to identify the stars which are entering or exiting the sensor frame. In this case all neighbor stars are considered to find which stars are entered to the current frame.

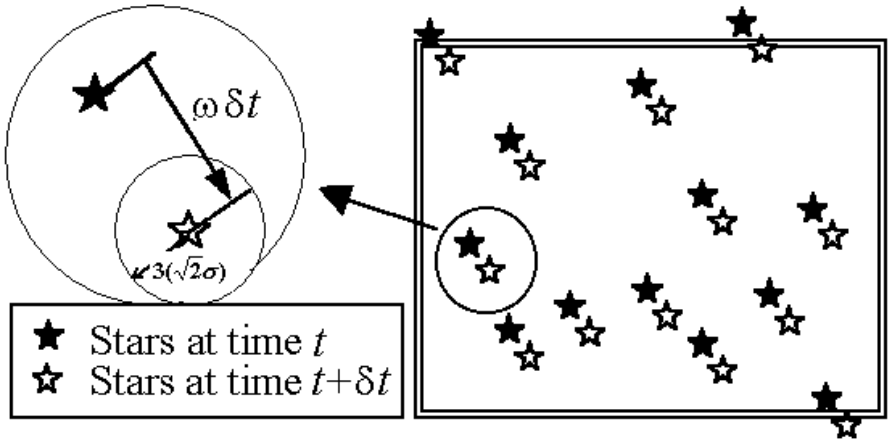

Figure 5.4 Stars image displacement due to camera motion

7. The interstar angle, between the unmatched stars and the star neighbors, is checked to find the ID for the unmatched stars.

The logic for the real time Star-ID is shown in Fig. 5.5.

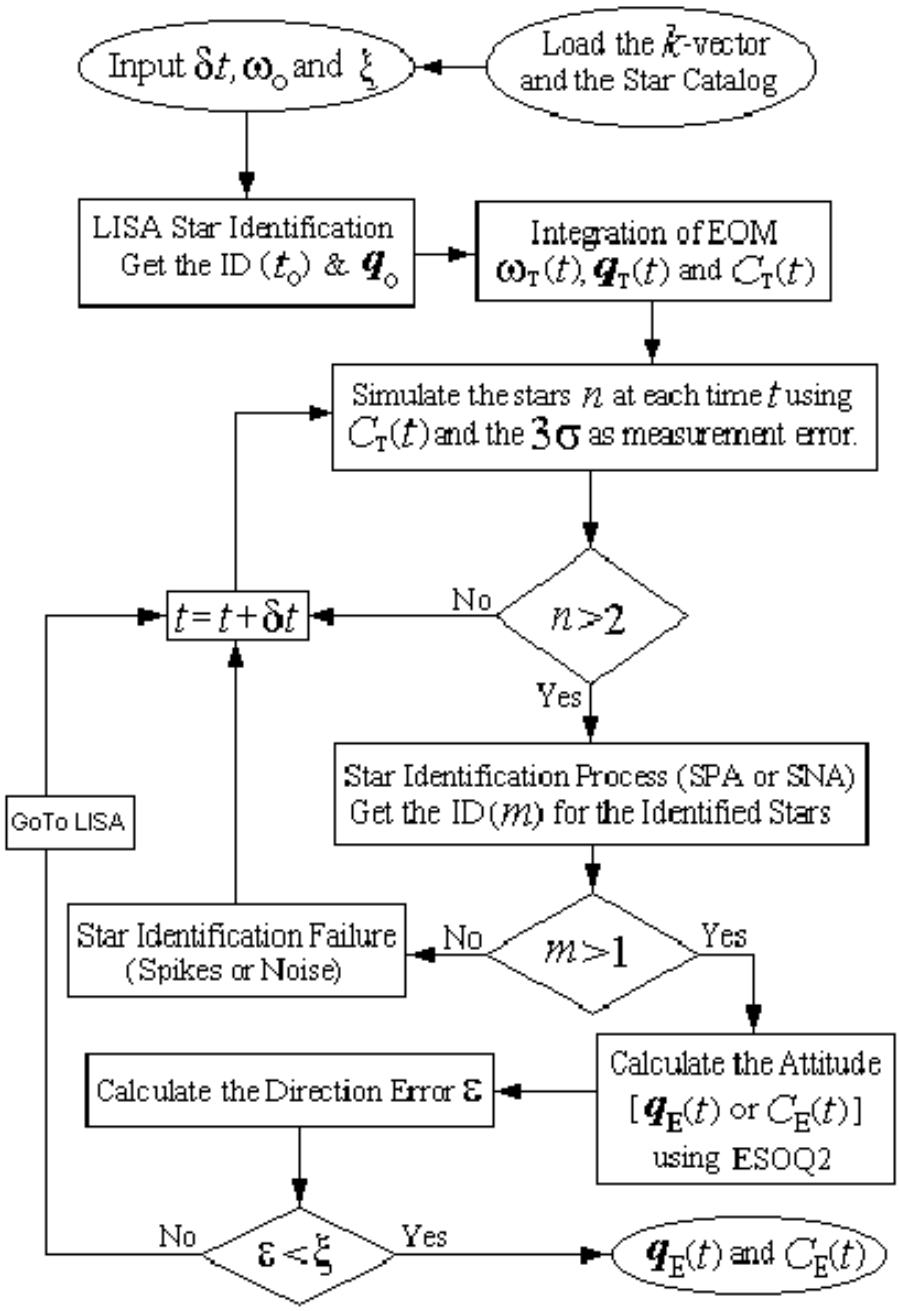

Figure 5.5 Logic flow diagram for the real time star identification program

5.4 Real Time Simulation Results

End to end numerical tests, based on simulation of true prescribed attitude motion, star image centroid measurements (including Gaussian measurement errors with zero mean and 17μ-radians (1-σ) standard deviation), star catalog access, Star-ID, and attitude estimation have been carried out. We report here a simulation that is based on a typical one minute of elapsed real time and using a time step (successive star image frame interval) of $\delta t = 0.01$ seconds. Firstly, the LISA program is called to obtain the solve the lost in space problem to estimate the initial attitude with no prior information, the output of this process is the optimal estimate for the quaternion $\mathbf{q}(t_0)$ and the associated direction cosine matrix $C(t_0)$ orienting the camera with respect to the inertial frame in which the cataloged star information is stored. Secondly, the estimated attitude dynamics is obtained by integrating, between successive times the quaternion kinematics equation $\dot{\mathbf{q}} = \frac{1}{2}\Xi(\omega)\mathbf{q}$ together with the angular velocity ω measurements provided by three rate gyros. Then, the star image observation data are simulated using the $C(t)$ and the true known star information from the catalog, together with random numbers to simulate centroiding errors of 17μ-radians (1-σ).

Figure 5.6 shows the histogram of the number star occurrence for the simulated star measurements. The Star-ID procedures are used to identify the stars using either the SP-Search or the SNA methods, and of course, the fact that the true attitude and Star-ID are known allows us to immediately observe any anomalies in the process.

The overall amount of the floating-point operations required are plotted in Fig. 5.7, as a function of the number of identified stars. All computations were done in MATLAB, and the FLOPS operator was utilized to obtain the floating-point operation count. As is well known, the FLOPS count is an imperfect indication of computational load, due the fact that some logical tests and integer operations are

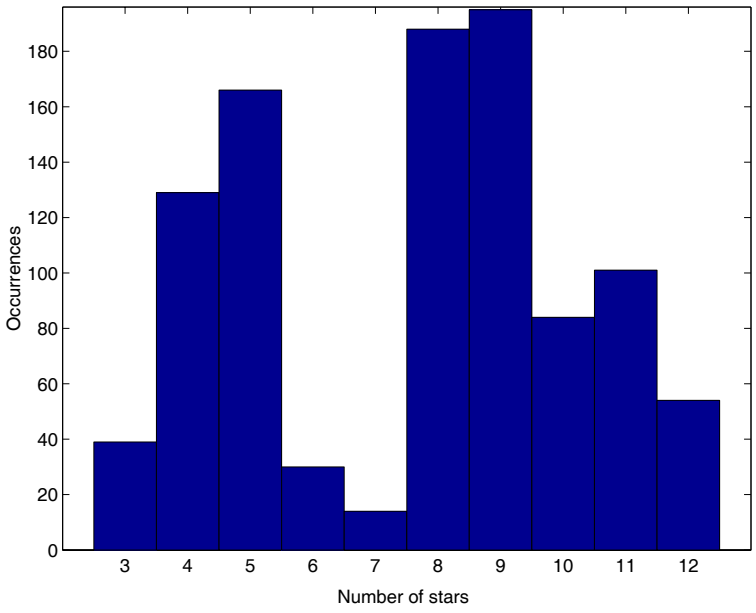

Figure 5.6 Histogram of the number of star occurrences

excluded by the FLOPS calculation. Therefore, the time consumed time, using a PC 450MHz operating under Windows 98, is also shown in Fig. 5.7. This plot gives a more meaningful measure of computations that of course can be reduced by over one order of magnitude by utilizing compiled code from, for example, equivalent C-code and an appropriate real time operating system.

Now, by using the fastest available attitude estimator (that is, ESOQ-1 [40] or ESOQ-2 [20], with the latest improvement [31]), the estimated attitude direction cosine matrix $C_E(t)$ is calculated using the observed star vectors and the cataloged star vectors of the identified stars. The parameter that describes the maximum "distance" between two orientation matrices (as, for instance, the true and the estimated attitude matrices), is described as the maximum direction error $\varepsilon_{\max}$, that is computed according to

$$\varepsilon_{\max} = \cos^{-1}\left\{[\text{trace}(C_T\, C_E^{\text{T}}) - 1]/2\right\} \tag{5.8}$$

now, since the true attitude C_T is always unknown - that implies that the principal

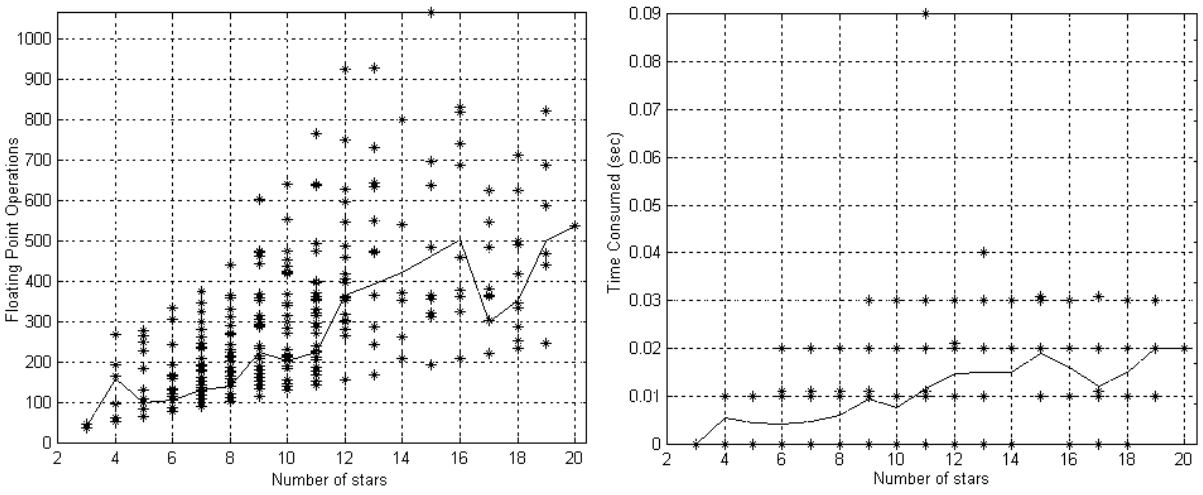

Figure 5.7 Floating point operations count and consumed time

axis direction of the attitude corrective matrix ($C_T\, C_E^{\mathrm{T}}$) is unknown - the "distance" between the true and the estimated attitude matrices, is better described by the expected value

$$E\{\varepsilon\} = \varepsilon_{\max}\, \pi\, /4 \tag{5.9}$$

Figure 5.8 shows the expected direction error $E\{\varepsilon\}$ as a function of the number of identified stars. This error is actually polarized about the direction of the optical axis $\mathbf{w}$. Therefore, the overall error, described by Eq. (5.8), can be decomposed in two parts: ε_a which is associated with the error of $\mathbf{w}$, and the remaining part ε_n orthogonal to it. Mathematically, these components are, therefore, computed using the following expressions

$$\begin{cases} \cos\varepsilon_a = \mathbf{w}^{\mathrm{T}}\mathbf{w}_{true} = \mathbf{w}^{\mathrm{T}}(\Delta\mathbf{w}) \\ \cos\varepsilon_n = \mathbf{n}^{\mathrm{T}}\mathbf{n}_{true} = \mathbf{n}^{\mathrm{T}}(\Delta\mathbf{n}) \end{cases} \tag{5.10}$$

where $\mathbf{n}$ indicates any random direction perpendicular to $\mathbf{w}$ and $\Delta = (C_T\, C_E^{\mathrm{T}})$. The numerical values of ε_a and ε_n, obtained by numerical tests, are shown in Fig. 5.9.

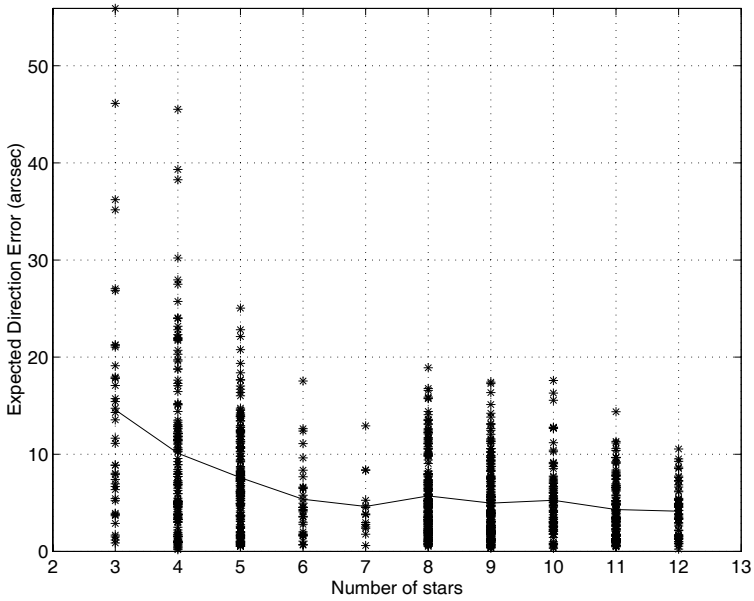

Figure 5.8 The expected direction error

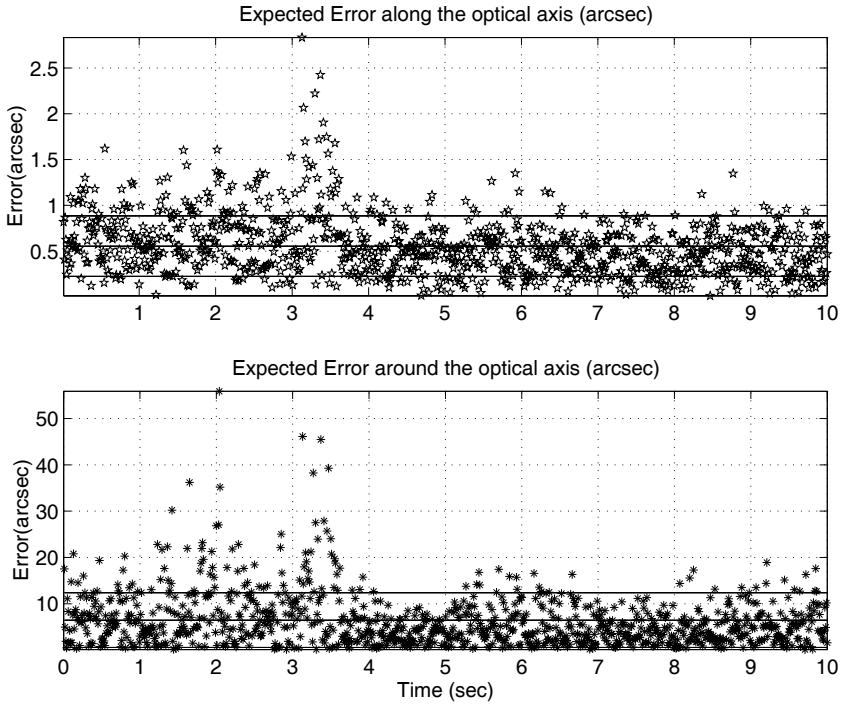

Figure 5.9 Expected errors (of and about) the optical axis

CHAPTER VI

NON-DIMENSIONAL STAR IDENTIFICATION FOR UNCALIBRATED STAR CAMERAS

6.1 Introduction

Star identification is an important process for any star tracker attitude determination sensor. The main purpose of the star identification is to identify the measured stars with the corresponding cataloged stars. Most star identification methods depend on the star camera parameters which are the focal length f and the focal plane offsets $(x_0,\ y_0)$. But in many cases these parameters may not be accurate or the camera may not yet be well calibrated. The Non-Dimensional Star-ID method is the perfect way to identify the stars of uncalibrated or poorly calibrated cameras. Of course this method also works for well calibrated star cameras. In this method we use the fact that the focal plane angles are independent, to first order, of both the *focal length as well as principal point offsets*. This method is very fast, easy to implement and accurate with a probability of failed Star-ID less than 10^{-11} for typical star tracker design parameters. For successful star identification, some camera parameters such as effective focal length and principal point offsets should be determined before any space mission. These parameters can be estimated in the laboratory or using a recently developed ground calibration algorithm [21]. However, due to some on-orbit distortion in the focal plane, or some thermal problems, these parameters may not remain accurate over the mission lifetime. Therefore, we are seeking a novel method that can determine the star identification for poorly calibrated cameras or when camera parameters are not accurate enough. The proposed method in this

chapter is called the "*Non-Dimensional Star-ID*" (the contents of this chapter are summarized from [24]).

We prove that the relation between the angles of the cataloged triangles (α_i) and the corresponding angles of focal plane measured triangles (β_i) are independent, to the first order, of the camera basic parameters (x_0, y_0, f). Their differences reflect primarily centroiding errors, and secondary, higher order distortions. For the typical case, usual manufacturing tolerance result in these higher order distortions being sufficiently small that the Non-Dimensional Star-ID method is very reliable. To verify our results, both night sky tests and Monte Carlo simulations is used to verify the Non-Dimensional Star-ID method.

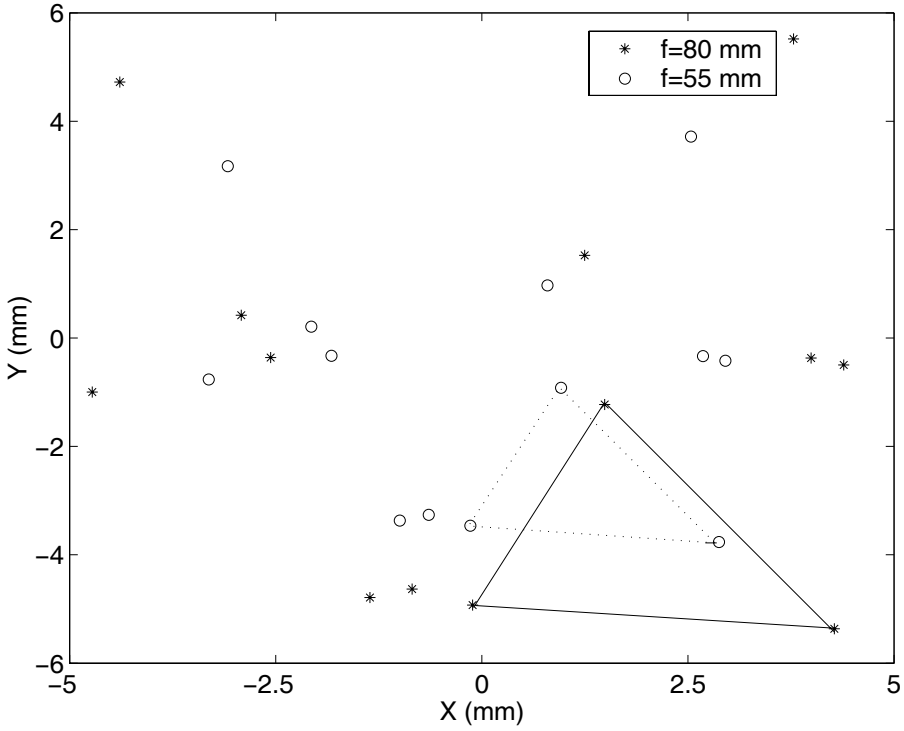

Figure 6.1 Star image for two different focal lengths

Figure 6.1 shows two star images for different focal lengths, and focal axis offsets. It is possible to demonstrate that, on a first approximation, the angles of focal plane

triangles are independent of the focal length, the rotation, and the offsets of the focal axis. The fact that variations in (x_0, y_0, f) simply shift and scale the triangles (to first order), means that using in-plane angles for Star-ID can likely be done with very poor estimates of (x_0, y_0, f). The Non-Dimensional Star-ID algorithm can also be used as the first step for the focal plane calibration to find camera basic parameters without any need for manual Star-ID.

6.2 Focal Plane and Inertial Angles

Assuming that the star tracker can be modelled as an ideal pinhole camera (see Fig. 6.2), the body vector (b_i) of any star measurement is related to the corresponding inertial unit-vector $\hat{r}_i$ through

$$b_i = \left\{ \begin{array}{c} -(x_i - x_0) \\ -(y_i - y_0) \\ f \end{array} \right\} = m_i \, \hat{b}_i = m_i \, A \, \hat{r}_i \tag{6.1}$$

where A is the attitude matrix, $m_i = \sqrt{(x_i - x_0)^2 + (y_i - y_0)^2 + f^2}$, f is the focal length, and (x_0, y_0) are the focal plane offsets. Equation (6.1) allows us to write

$$\left\{ \begin{array}{l} \Delta b_{ij} = b_j - b_i = \left\{ \begin{array}{c} x_i - x_j \\ y_i - y_j \\ 0 \end{array} \right\} = A(m_j \hat{r}_j - m_i \hat{r}_i) \\ \\ \Delta b_{ik} = b_k - b_i = \left\{ \begin{array}{c} x_i - x_k \\ y_i - y_k \\ 0 \end{array} \right\} = A(m_k \hat{r}_k - m_i \hat{r}_i) \end{array} \right. \tag{6.2}$$

The angle α is simply evaluated from the reference frame vectors by

$$\cos \alpha = \frac{(\Delta r_{ij})^T \, (\Delta r_{ik})}{\|\Delta r_{ij}\| \, \|\Delta r_{ik}\|} \tag{6.3}$$

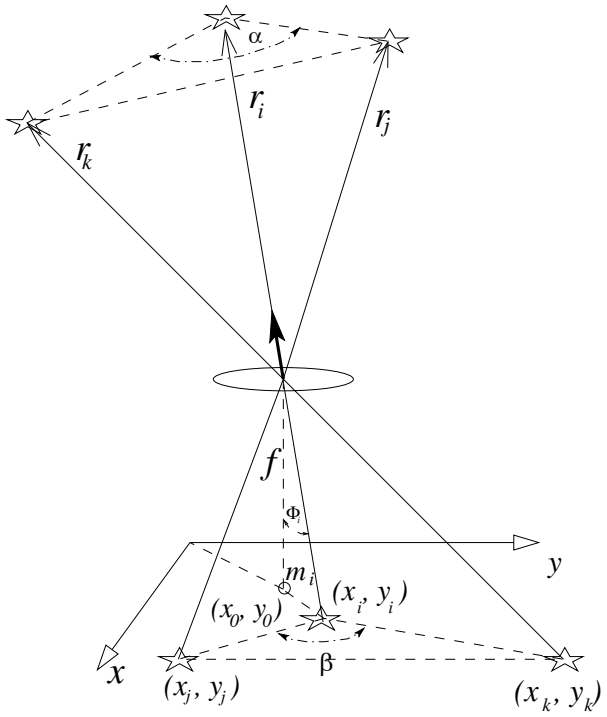

Figure 6.2 Geometry for the pin-hole camera (co-linearity condition)

Also, assuming the usual pinhole camera model, the corresponding focal plane angle is evaluated as

$$\cos\beta = \frac{(\Delta b_{ij})^T(\Delta b_{ik})}{\|\Delta b_{ij}\|\|\Delta b_{ik}\|} = \frac{(x_j - x_i)(x_k - x_i) + (y_j - y_i)(y_k - y_i)}{\sqrt{(x_j - x_i)^2 + (y_j - y_i)^2}\sqrt{(x_k - x_i)^2 + (y_k - y_i)^2}} \tag{6.4}$$

Now, the vector magnitude may be expressed as

$$m_i = f(1 + \delta_i) \qquad \Longleftrightarrow \qquad \delta_i = \frac{m_i - f}{f} \tag{6.5}$$

where $i = j, k$. Now, for a 10° pine-hole camera ($\cos\phi_i > 0.9962$ and $f = m_i \cos\phi_i$), we have the upper limit $\delta_i = (\cos\phi_i)^{-1} - 1 < 0.0038$. Therefore, Eqs. (6.2) become

$$\begin{cases} \Delta b_{ij} = f\,A\,[\,(1+\delta_j)\,r_j - (1+\delta_i)\,r_i\,] = f\,A\,[\,r_j - r_i + \delta_j\,r_j - \delta_i\,r_i\,] \\ \Delta b_{ik} = f\,A\,[\,(1+\delta_k)\,r_k - (1+\delta_i)\,r_i\,] = f\,A\,[\,r_k - r_i + \delta_k\,r_k - \delta_i\,r_i\,] \end{cases} \tag{6.6}$$

Notice that the last two terms in Eqs. (6.6) are very small < 0.0038. Substituting

Eqs. (6.6) into Eq. (6.4), and using $A^T A = I$, we obtain

$$\cos\beta = \frac{(r_j - r_i)^T(r_k - r_i) + N + \varepsilon}{D_1\, D_2} \tag{6.7}$$

where,

$$\begin{cases} N = \delta_j r_j^T(r_k - r_i) - \delta_i r_i^T(r_k - r_i) + \delta_k r_k^T(r_j - r_i) - \delta_i r_i^T(r_j - r_i) \\ D_1 = \sqrt{(r_j - r_i)^T(r_k - r_i) + 2(\delta_j r_j - \delta_i r_i)^T(r_j - r_i)} \\ D_2 = \sqrt{(r_k - r_i)^T(r_k - r_i) + 2(\delta_k r_k - \delta_i r_i)^T(r_k - r_i)} \\ \varepsilon = \delta_j\delta_k r_j^T r_k - \delta_j\delta_i r_i^T r_j - \delta_i\delta_k r_i^T r_k + \delta_i^2 \end{cases} \tag{6.8}$$

For negligible $(\delta_i,\ \delta_j,\ \delta_k)$, Eq. (6.7) is identical to Eq. (6.3) and then $\beta = \alpha$. Notice that $\cos\beta$ depends on $(x_0,\ y_0, f)$ only through the small $(\delta_i,\ \delta_j,\ \delta_k)$ terms. Since the δ_i terms are all at worst of the order ~ 0.0038. A small error in $(x_0,\ y_0,\ f)$ perturbs the already small terms, making sensitivity w.r.t. $(x_0,\ y_0,\ f)$ errors very low. For example, look at the variation of the δ_i term due to an error Δf in focal length.

$$\Delta\delta_i = \frac{\partial\delta_i}{\partial f}\Delta f = \left(\frac{\frac{\partial m_i}{\partial f} - 1}{f} - \frac{m_i - f}{f^2}\right)\Delta f = \left(\frac{\frac{1}{\cos\phi_i} - 1}{f} - \frac{\frac{m_i}{f} - 1}{f}\right)\Delta f \tag{6.9}$$

For a typical example if f =60 mm and $\Delta f = 30$ mm (which is very large error in focal length!) we can get $\Delta\delta_i < 10^{-4}$. For a more reasonable 1 mm error in f, we find $\Delta\delta_i < 10^{-6}$ which is smaller than centroiding errors of modern star trackers!

So, the worst case variation in δ_i is at least one order of magnitude smaller than δ_i. This indicates that $\cos\beta$ has a very low sensitivity to any reasonable errors in f. This error in the $\cos\beta$ is typically less than the tolerance (associated with camera accuracy) for our search. Even when we ignore δ_i corrections, we can achieve reliable star identification by considering redundant measured stars, as shown in the next section.

6.3 The Non-Dimensional Star-ID

From Fig. 6.2, the cataloged vectors and the focal plane angles are calculated as follow ($r_{ij} = r_j - r_i$, $r_{ik} = r_k - r_i$, and $r_{kj} = r_j - r_k$)

$$\cos\alpha_1 = \frac{r_{ij}^T r_{ik}}{\|r_{ik}\|\|r_{ik}\|}, \qquad \cos\alpha_2 = \frac{-r_{jk}^T r_{ij}}{\|r_{jk}\|\|r_{ij}\|}, \qquad \cos\alpha_3 = \frac{r_{ik}^T r_{jk}}{\|r_{ik}\|\|r_{jk}\|} \tag{6.10}$$

The indices of the cataloged vectors and the maximum and minimum angles of the focal plane triangles are stored in a matrix. The size of this matrix depends on the magnitude threshold, i.e for $M_{th} = 5.0$ the number of admissible triangles for FOV = 8 deg. is 55,309 triangles. For $M_{th} = 5.5$ the number of admissible triangles for FOV = 8 deg. is 338,369 triangles. Figure 6.3 shows a plot of the smallest angle (α_3) versus the focal plane triangle index (k). For the sake of illustration, we assume no prior knowledge of (x_0, y_0, f) and ignore the δ_i and δ_i^2 corrections in Eq. (6.10). We adopt a worst case tolerance for angle matching of 0.005 rad corresponding to the maximum δ_i possible over FOV.

If we divide the smallest angle (α_3) into a very small and uniform interval ($\delta\alpha_3$) and interpolate the value of (k) at any value of ($0^\circ < \alpha_3 < 60^\circ$). So, for ($\delta\alpha_3 = 0.001^\circ$) we can store the values of α_3 and the corresponding index value k. Figure 6.4 illustrates this idea. Even though Fig. 6.3 looks smooth, there is evident fine structure variations on the finer scale of Fig. 6.4. The uniform in (α_3) table, a portion of which plotted in Fig. 6.4 is introduced to permit ultra high speed, search-less interpolation of the index k as a function of measured values of α_3. Table 6.1 shows a portion of the star triangle data sorted with respect to α_3.

Now, once we have the coordinates of any star image ($x_i,\ y_i$) we can find the focal plane angles established by any three stars. The smallest angle of each triangle

Table 6.1 A portion of the star triangle data

k	I_1	I_2	I_3	α_1	α_3
11453	3903	1187	2230	102.1256	6.022182
11454	2434	725	2187	143.2689	6.022355
11455	729	3787	4722	101.0456	6.022963
11456	3487	1291	2229	130.1984	6.023178
11457	2128	3315	2641	147.359	6.023315
11458	483	2578	1424	147.5892	6.023319
11459	781	2895	3232	139.8646	6.023402
11460	4723	514	4195	141.9004	6.023447
11461	1315	667	3047	146.2288	6.023514
11462	632	2737	4757	118.9169	6.023539
11463	3684	1542	503	146.5031	6.023613

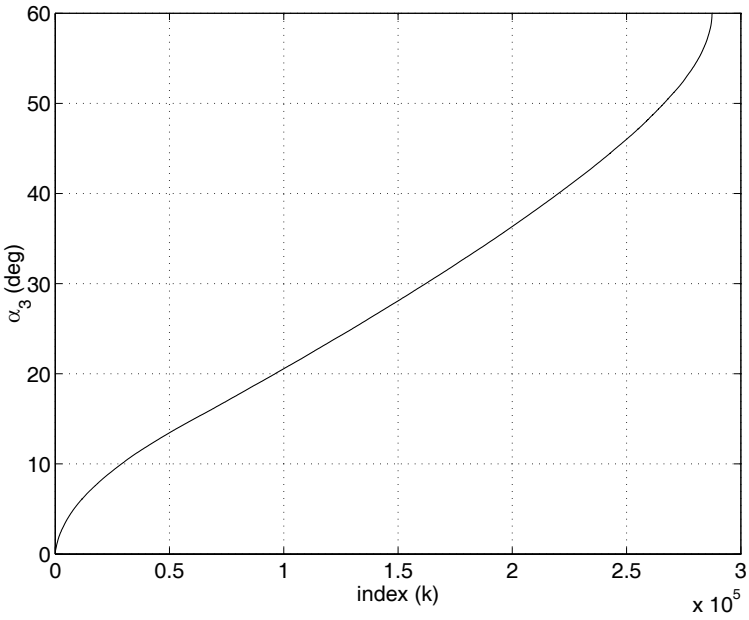

Figure 6.3 The smallest focal plane angle versus its index

is used to find, without a search, the index range (k_l to k_u) using

$$k_l = \text{floor}\left(\frac{\beta_3 - \beta_{3error}}{\delta\beta_3}\right) \quad \text{and} \quad k_u = \text{ceil}\left(\frac{\beta_3 + \beta_{3error}}{\delta\beta_3}\right) \tag{6.11}$$

using this range and by accessing the stored matrix for the cataloged indices we can find the range of the candidate triangles. By checking the largest angle β_1 we can find exactly which triangle in the catalogue corresponds to measured one. As mentioned, we use a tolerance of $\beta_{3error} = 500\mu rad$ corresponding to the maximum error in using Eq. (6.10) to approximate Eq. (6.7). For this crude tolerance, we may occasionally fail if 3 or 4 stars are matched, but adding a 5^{th} star results in a virtually certain match.

Table 6.1 contains entries: $K,\ \alpha_3(k),\ \alpha_1(k),\ I_1(k),\ I_2(k),\ I_3(k)$, where $I_i(k)$ are the three stars indices for the K^{th} cataloged triangle. The master catalog contains entries $I,\ M_v(I),\ \lambda(I),\ \mu(I)$, where I is the Star-ID integer, $M_v(I)$ is the visual magnitude, $\lambda(I)$ is the right ascension and $\mu(I)$ is the declination. If more than one

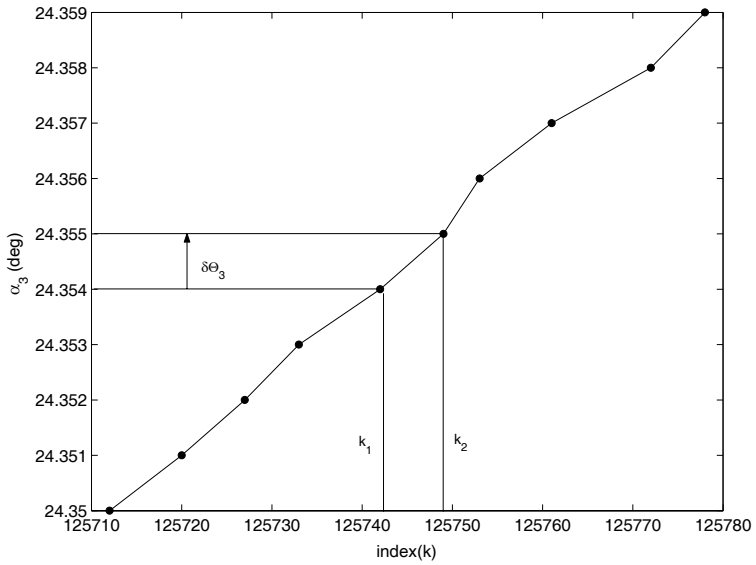

Figure 6.4 α_3 versus its index

triangle is found we exclude this one and form another triangle till we find a unique matched triangle.

If a unique triangle is found (most common case), the Star-ID can be confirmed, to very high confidence by using 4^{th} and 5^{th} stars to form new triangles sharing two stars with the first matched triangle. Finally, all interstar angles of the polygon must match to within the tolerance (6 for a 4 star pattern and 10 for a 5 star pattern). When these second triangles are matched, of course the indices of the two common stars must also match; otherwise, the initial triangle match is rejected. When this occurs (usual case), the frequency of mismatch of four stars is on the order of 10^{-11} (for magnitude 5.5 threshold, 8° FOV, 512×512 pixel format).

6.4 Non-Dimensional Star-ID Algorithm Results

The above method is programmed using MATLAB and it has proven very successful for night sky tests and also for Monte Carlo star image simulations.

6.4.1 Using Night Sky Tests

Using the StarNav I prototype camera Pegasus 512×512 [41], we performed a night sky test. Figure 6.5 shows an actual star image centered near $\alpha = 92°, \delta = 31°$. Figure 6.6 shows the coordinates of the star image done by the centroiding algorithm [16]. Table 6.2 shows the results of the star coordinates.

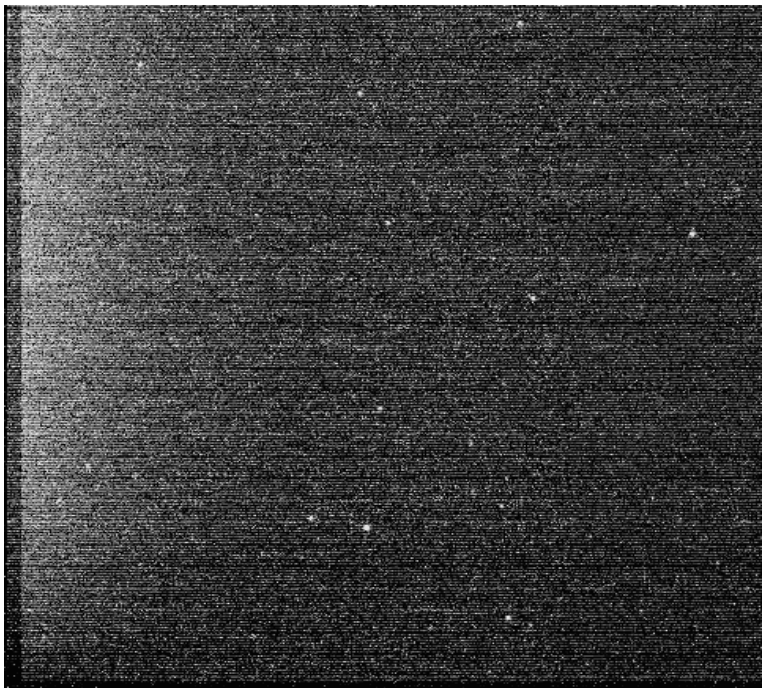

Figure 6.5 Star image using the Pegasus camera

Table 6.2 The star coordinates for Jupiter

X	464.29	244.54	340.56	357.17	207.18	91.55
Y	339.96	119.37	51.8	291.92	127.14	466.47
X + error	464.2422	244.1238	340.7072	356.5019	207.5372	92.3618
Y + error	339.6141	119.799	52.4270	291.1231	126.4195	466.7556

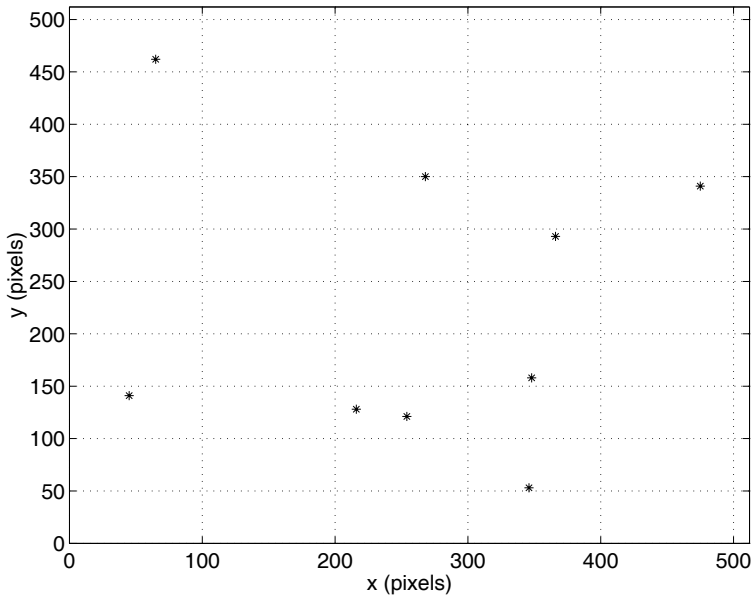

Figure 6.6 The centroiding results for the Pegasus image

After applying the Non-Dimensional Star-ID, the results are shown in Table 6.3.

Table 6.3 Star-ID results for Jupiter

Star-ID	669	1655	2546	1825	2545	2019
R.A.	−86.1554	−91.6681	−90.5015	−88.4066	−92.2581	−91.8330
Decl.	29.4983	27.6122	25.9538	29.5125	27.9679	33.9175

Also, by using the least square algorithm to find the bore-sight and focal length [21] we get the calibrated focal length $f_c = 52.3292$ mm, the calibrated y-axis offset $x_0 = -0.28053$ mm, and the calibrated x-axis offset $y_0 = -0.20124$ mm, as shown in Fig. 6.7.

Also, by using the GIFTS prototype camera Star1000 1024×1024 pixels, we performed a night sky test.

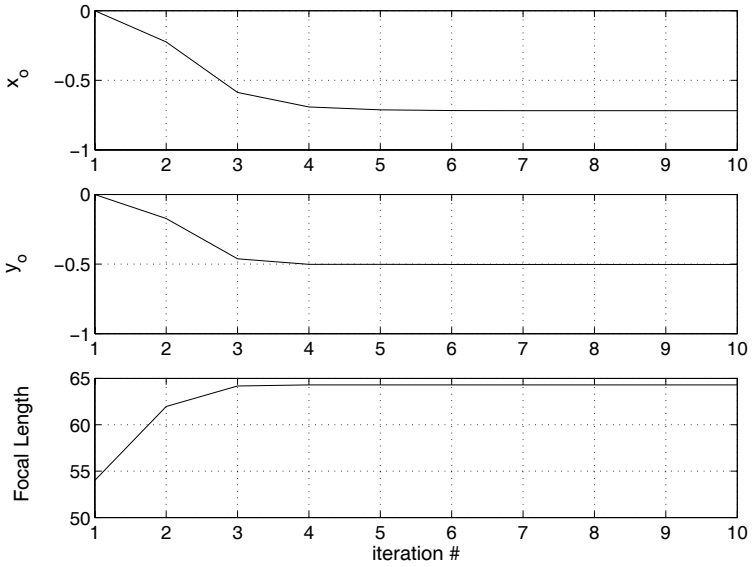

Figure 6.7 The calibrated camera focal length and offsets

Figure 6.8 shows an actual night sky star image using Star1000 camera.

Figure 6.9 shows the nine bright stars coordinates of the star image done by the centroiding algorithm [16].

After applying the Non-Dimensional Star-ID algorithm for the above star image we obtain the results in Table 6.4, which validates the manual Star-ID results.

Table 6.4 Star-ID results for Star1000

ID	1251	1255	945	936	1250	947	940	1249	1257
R.A.	42.5375	50.1292	49.7292	56.1250	42.0208	44.812	45.673	40.904	50.595
Dec	27.2739	29.0603	34.234	35.073	29.260	32.298	35.196	27.436	27.720

6.4.2 Using Monte-Carlo Simulations

MATLAB simulations have been performed to verify the above results. 1000 random tests, for various random spacecraft attitudes, have been done using a program which

Figure 6.8 Star image using Star1000 camera

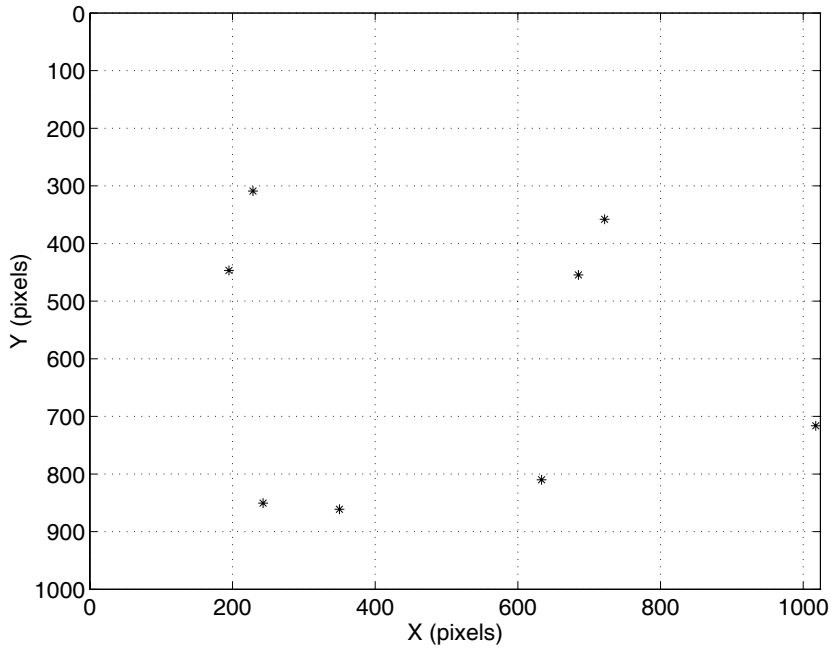

Figure 6.9 The centroiding results for the Star1000 image

simulates star images with Gaussian random centroiding errors. Figure 6.10 shows the resulting histogram of the number of observed star occurrences. In figure 6.11 the overall time required to perform the Non-Dimensional Star-ID, is shown.

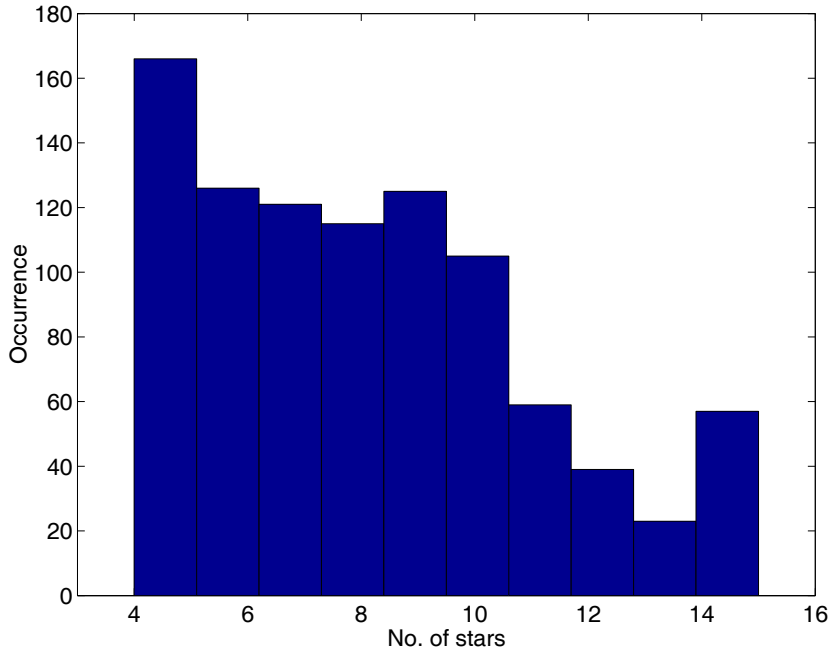

Figure 6.10 The histogram for the number of stars

For the 1000 tests we found that the empirical frequency of wrong Star-ID = 0%, the actual probability is approximately 10^{-11}. The only case that this Star-ID method fails is when there are too few valid imaged stars to form a triangle, or the stars lie in almost one line (in practice this case may happen with probability less than 1%) depending largely on integration time and catalog construction. An alternative logic can easily be developed to cover the small number of special cases; for attitude estimation in conjunction with rate gyros, an occasional "drop out" due to sparse star fields is easily tolerated. The execution times shown in Fig. 6.11 are using a 600 Mhz Pentium III PC. They depend of course upon the details of computing implementation, but nonetheless they provide an obvious basis for optimism. Algorithms based

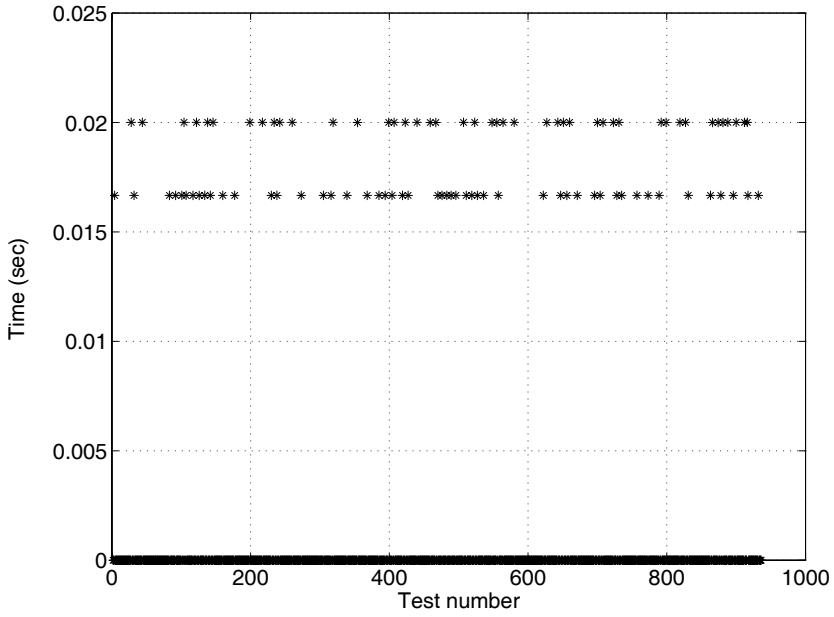

Figure 6.11 The execution time for each test

upon the *Pyramid* LISA algorithm [18], [19] are much more faster, however, they require a well calibrated camera. Obviously after the first stars are identified, the camera can be calibrated using the method presented in [21], the *Pyramid* algorithm can be used to identify the stars and virtually guarantee a correct result.

6.5 Analytical Solution for an Approximate Focal Length

After a set of stars are identified by the proposed Non-Dimensional approach, a good estimate of the focal length can be obtained as explained in this section. This is an important task that is needed to initiate faster Star-ID algorithms as, for instance, the *Pyramid* LISA. Hereafter, the procedure to estimate the f, is summarized.

(1) From Eq. (6.1), if we assume $(x_0,\ y_0)$ are negligible with respect to f, then we

can write the interstar angle of the measured stars as

$$\cos\phi_{ij} = b_i^T b_j = \frac{x_i x_j + y_i y_j + f^2}{\sqrt{x_i^2 + y_i^2 + f^2}\sqrt{x_j^2 + y_j^2 + f^2}} \tag{6.12}$$

Note the difference between this equation, which calculates the interstar angle ϕ_{ij}, and Eq. (6.4) that calculates the focal plane angle.

(2) The measured cosines should ideally equal the cataloged cosines, $\cos\phi_{ij}$, known from dot product of cataloged vectors. So we can form the following equation in which only f is unknown

$$\cos\phi_{ij}^2\,[\,(x_i^2 + y_i^2 + f^2)(x_j^2 + y_j^2 + f^2)\,] = (x_i x_j + y_i y_j + f^2)^2 \tag{6.13}$$

(3) If we re-arrange Eq. (6.13) to the quadratic polynomial, then the roots of this polynomial are the solutions for the focal length f

$$f^2 = \frac{(a+d)d^2 - 2c \pm \sqrt{[\,(a+b)d^2 - 2c\,]^2 - 4(1-d^2)(c^2 - abd^2)}}{2(1-d^2)} \tag{6.14}$$

where $a = x_i^2 + y_i^2$, $b = x_j^2 + y_j^2$, $c = x_i x_j + y_i y_j$, and $d = \cos\phi_{ij}$.

(4) Now, let us check the sign of $(c^2 - abd^2)$, for small field of view ($d^2 \cong 1$), then

$$\begin{aligned} c^2 - abd^2 = &\ (x_i x_j + y_i y_j)^2 - (x_i^2 + y_i^2)(x_j^2 + y_j^2) \cong \\ \cong &\ -[x_i^2 y_j^2 + y_i^2 x_j^2 - 2x_i x_j y_i y_j] \cong -[x_i y_j - x_j y_i]^2 \leq 0 \end{aligned}$$

(5) Since the sign of $[4(1-d^2)(c^2 - abd^2)]$ is negative, then the value of the square root is greater than $[(a+d)d^2 - 2c]$. By choosing the positive sign of the square root (because f^2 must be positive), then we have an algebraic solution for f.

Having this solution in hand, we can estimate the focal length from any pair of stars or average it if more known stars are available.

The above analytical result is tested using night sky images and simulated images and accurate values for the focal lengths are obtained. Although, we assume in our analysis that the focal plane offsets (x_0, y_0) are negligible but we test this analytical solution for large specified offsets ($x_0 = -0.2$ mm, and $y_0 = 0.25$ mm). For a random spacecraft attitude and a known camera focal length $f = 55$ mm, we calculate the star coordinates and identification to be as shown in Table 6.5. Each star pair is used to

Table 6.5 Star image coordinates and identification

X	2.2264	1.6275	2.5752	2.821	−0.305	1.8155	−1.9219	0.2891
Y	0.8718	2.0876	3.4253	2.195	3.3359	0.5425	−2.6363	−3.095
ID	408	677	678	1216	1331	1675	2304	2583

find the focal length using Eq. (6.14). Figure 6.12 shows the plot of the focal length for each pair. The mean value of the focal length is found to be 55.0029 mm and the standard deviation is 0.0171. Thus, it is not surprising that f is relatively insensitive to (x_0, y_0) errors. We have found that an approximate f computed using Eq. (6.14), along with starting estimates of $(x_0 \simeq 0,\ y_0 \simeq 0)$ are sufficient to ensure convergence of the nonlinear least squares algorithm [21], to obtain the final calibration estimates $(\hat{x}_0, \hat{y}_0, \hat{f})$.

Thus the sequence of events (for a poorly calibrated camera) might be as follows:

- Use the Non Dimensional Star-ID method to identify the stars,
- Use Eq. (6.14) to approximate the focal length f,
- Refine $(\hat{x}_0, \hat{y}_0, \hat{f})$ using the method in [21], and
- Use the *Pyramid* LISA method of [18] to confirm the identification, and for all subsequent star identifications.

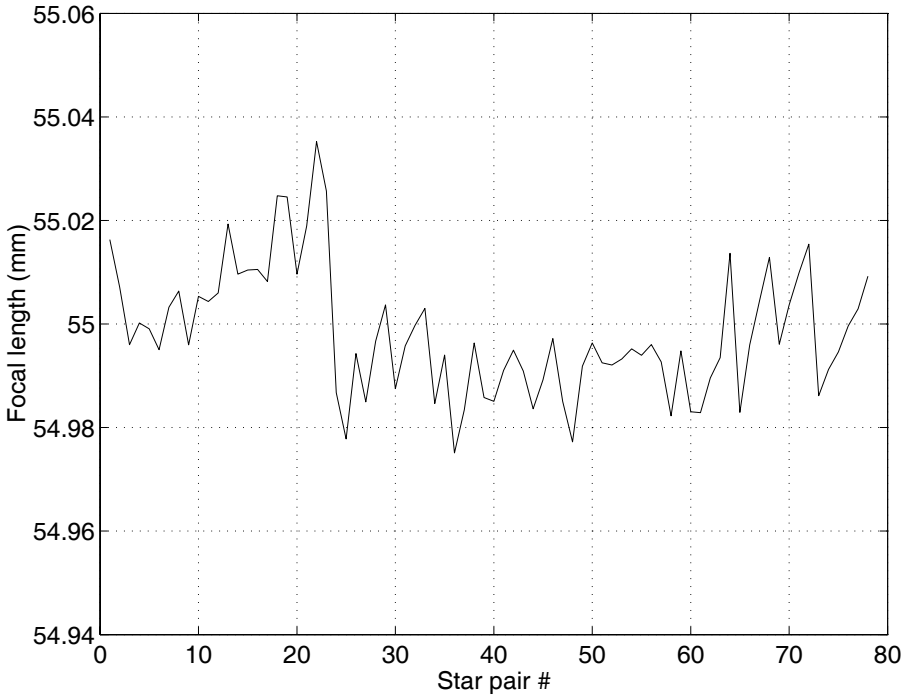

Figure 6.12 The approximate value for the focal length

CHAPTER VII

THE EFFECTS OF IMAGE SMEAR

7.1 Problem Description

Image smear refers to the situation when the imaged star is non-ideal, that is, when the imaged star is affected by the motion of the spacecraft. Image smear occurs in the focal plane of the imaged stars for slow values of the integration time t_s, or for high values of the spacecraft angular velocity ω. Interestingly, the algorithms presented in chapter (2), for the centroiding elliptical PSFs are applicable, to a degree, to the image smear problem (see [8] and [9]). Figure 7.1 shows a night sky star image for a randomly oriented star tracker with a 657 x 495 pixel focal plane detector and 55 mm focal length ST-237 camera, the exposure time (t_s) for this image is 1.0 second. A high precision telescope mount is used to perform an angular rotation of the camera around a certain axis of rotation. Figure 7.2 shows the same star image as in Fig. 7.1 with image smear $\omega = 1/12$ deg/sec and for $t_s = 0.5$ sec. Figure 7.3 shows the star image with image smear $\omega = 0.5$ deg/sec and for $t_s = 0.5$ sec. Figure 7.4 shows the star image with image smear $\omega = 0.5$ deg/sec and for $t_s = 2.0$ sec. These night sky images validate the fact that the volume of the intensity distribution of each star is approximately constant during the integration time while the significantly illuminated area increases with increasing the smear.

These qualitative remarks break down at higher slew rates, when the accumulated energy becomes comparable to pixel dark current noise. We note that implementing the split field of view, with astigmatism tagging, presents a difficulty if there is significant image smear, because it is difficult to distinguish between smear and astig-

matism. As a consequence, with the astigmatism we restrict the angular velocity to be below a thresh-hold value which results in negligible image smear. But more generally, some of the image processing ideas for astigmatic elliptical star images can be used to discern image data from smeared images.

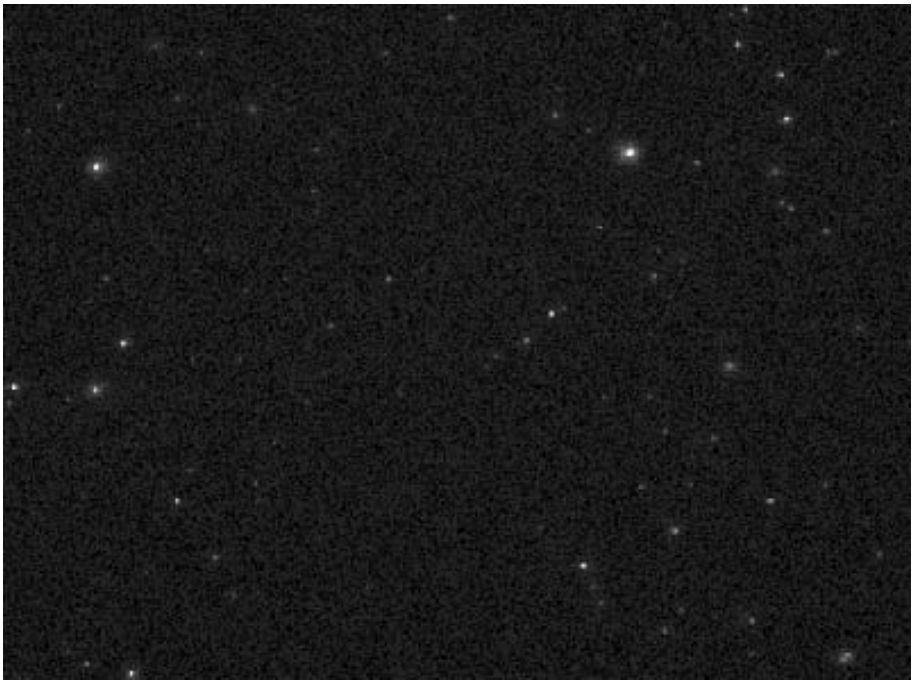

Figure 7.1 Star image without smear at 1.0 sec exposure time

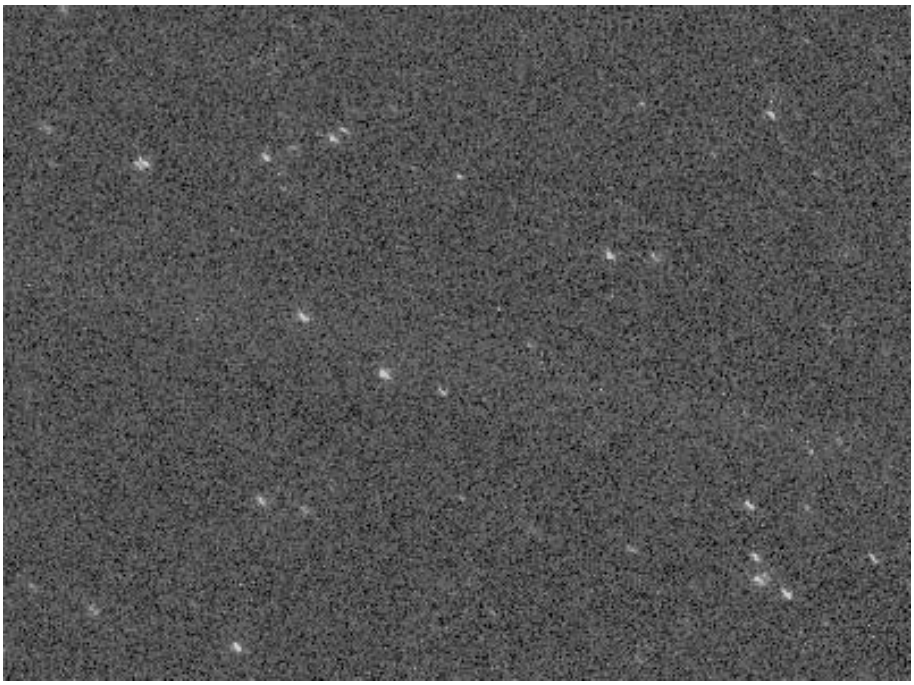

Figure 7.2 Star image at 0.5 sec exposure and smear = 1/12 deg/sec

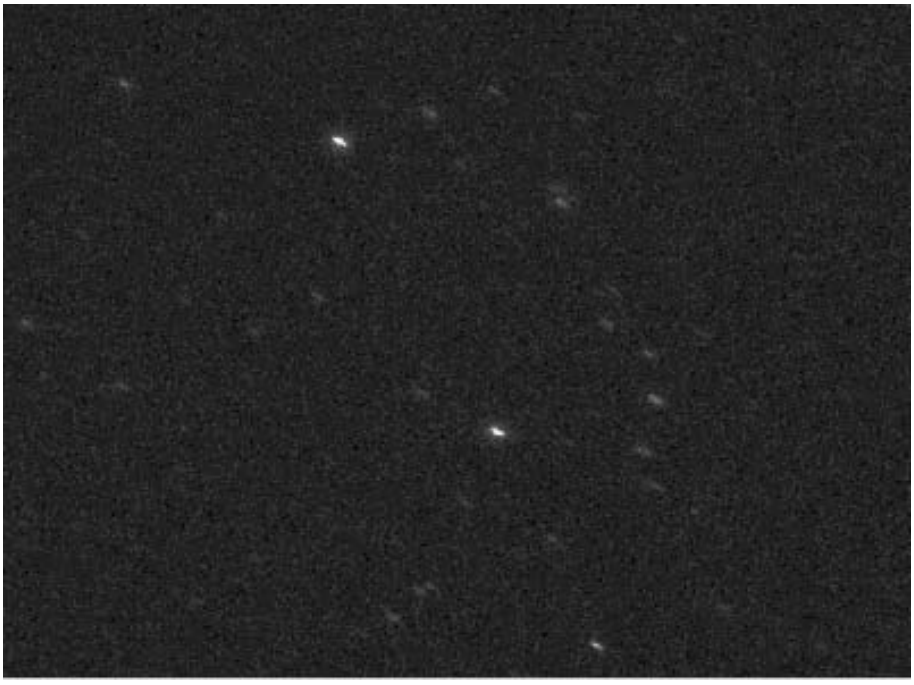

Figure 7.3 Star image at 0.5 sec exposure and smear = 0.5 deg/sec

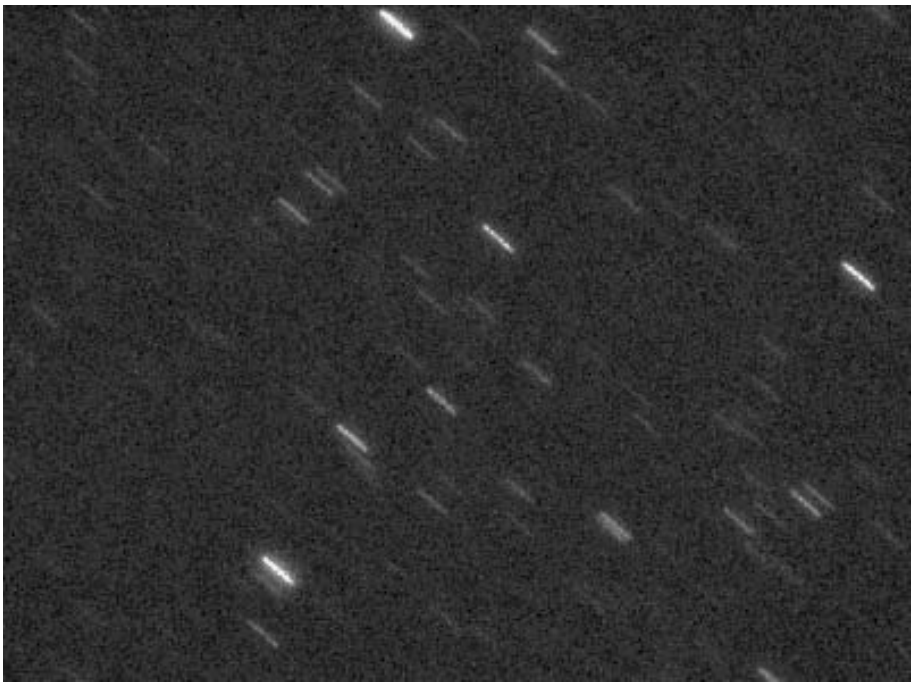

Figure 7.4 Star image at 2.0 sec exposure and smear = 0.5 deg/sec

7.2 Maximum Angular Rate Estimation

The centroiding techniques allow the star direction to be determined with a precision of 1/10 of a pixel or better; this conservative estimate is derived from night sky experiments. From Fig. 7.5 we can calculate the corresponding maximum allowed

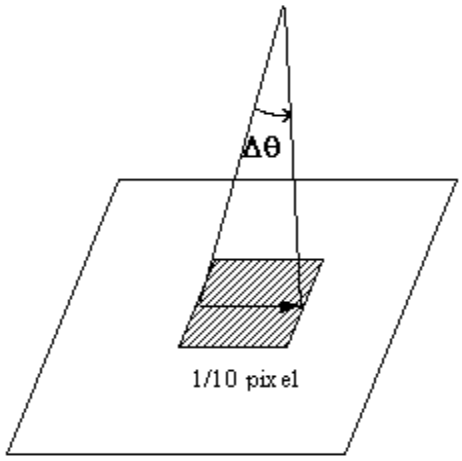

Figure 7.5 Centroiding accuracy

separation angle for totally negligible image smear as $\Delta\vartheta = 1/10$ (chip size/number of pixels). The critical condition $\Delta\vartheta = \omega\delta t$, where $\delta t \geq t_s$ is the time between successive images and ω is the angular rate of the spacecraft corresponds to image smear $\Delta\vartheta$ being equal to the random, approximately Gaussian centroiding errors. The analytical values of the maximum angular velocity, for negligible image smear as a function of integration time are calculated for a typical 512×512 CCD camera and a 7^o field of view. Table 7.1 summarizes the results of the analytical value of the maximum angular velocity for each integration time.

For $\omega < \omega_{max} = \Delta\vartheta/\delta t$ results in negligible image smear (smaller than expected centroiding error), however, we find in practice that acceptable centroid accuracy (50 μ rad) may be obtained, for perhaps two or three times these "max" angular velocity

Table 7.1 Analytical maximum angular velocity

t_s (msec)	10	20	30	50
$\omega_{\max}$ (deg/sec)	0.156	0.078	0.052	0.0313

estimates, this is discussed in the next section.

7.3 Measurement Errors and the Standard Deviation Computation

For the star images, taken with smear effect, the attitude associated with the image should be most valid at $(t+\delta t/2)$. For simplicity, in the present discussion we consider $t_s = \delta t$. So, if no smear the $\Delta\vartheta = \omega\delta t$ is negligible and the attitude matrix is $C(t)$. For the smear case the attitude matrix at $(t + \delta t/2)$ is

$$C(t + \delta t/2) = [I - \tilde{\omega}\delta t/2]\, C(t) \tag{7.1}$$

The equivalent angular centroiding (measurement) error is estimated from a finite sample (N) as

$$\sigma^2 = \frac{1}{N} \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} (\vartheta_{ij}^M - \vartheta_{ij}^C)^2 \tag{7.2}$$

where ϑ_{ij}^M is the interstar angle measured from smeared image, ϑ_{ij}^C is the corresponding interstar angle from cataloged vectors, $N = n(n-1)/2$ is the number of star pairs, and n is the number of measured stars. The night sky images for the star images with and without smear are used to calculate the measurement errors, for different angular velocities and integration times. Notice σ^2 can be computed from measured and cataloged interstar angles without using an attitude estimate.

Table 7.2 shows the effect of the image smear on the centroiding errors for integration times equal to 10, 20 and 30 msec, respectively. This table is created using

Table 7.2 Measurement errors and standard deviations for various cases

$t_s = 10$ msec	ω (deg/sec)	0.01	0.02	0.05	0.10	0.20	0.25	0.30
$t_s = 10$ msec	$\sigma_{w/o}$ (μ rad)	35.6	-	-	-	-	-	35.6
$t_s = 10$ msec	σ_{smear}	36.0	36.5	36.7	37.5	56.4	60.6	105.0
$t_s = 10$ msec	$\text{STD}_{w/o}$	39.1	-	-	-	-	-	39.1
$t_s = 10$ msec	STD_{smear}	39.1	39.2	39.6	41.2	51.7	57.0	97.0
$t_s = 20$ msec	ω	0.01	0.02	0.05	0.10	0.15	0.20	0.25
$t_s = 20$ msec	σ_{smear}	35.6	35.9	37.4	56.8	168.0	246.0	429.0
$t_s = 20$ msec	STD_{smear}	39.7	42.0	44.6	52.4	54.4	228.0	424.0
$t_s = 30$ msec	ω	0.01	0.02	0.05	0.10	0.15	0.20	0.25
$t_s = 30$ msec	σ_{smear}	37.0	38.2	45.5	81.9	292.4	521.0	808.2
$t_s = 30$ msec	STD_{smear}	39.5	42.8	47.7	76.0	284.8	485.8	773.0

a finite number of night sky images, but is believed to be converged to within 1% or better. Notice that accuracy degrades slowly with ω, until $\omega \cong 2\omega_{max}$, see table (7.2).

As expected the measurement errors get worse with increasing angular velocity (smear). So, by comparing Table 7.1 with Table 7.2 we can conclude that, the actual angular velocity can be increased by factor of about two to three from the analytical maximum angular velocity estimate in Table 7.1, with small to moderate accuracy degradation.

CHAPTER VIII

STAR IDENTIFICATION RELIABILITY

8.1 Introduction

Design of the spacecraft attitude may be dependent on the probability of a successful autonomous acquisition of the stellar attitude determination system. The problem of real-time, on-board star pattern identification (studied in Chapter IV) which must precede any spacecraft attitude estimation algorithm based upon measured line of sight directions to stars, is further studied in this chapter. Following the use of the searchless *k-vector* method to access the feasible candidate stars for each measured pair; a tiered logical structure is introduced where interstar angles for pairs, triples and quadruples, and generalized elementary polygons are used to match measured star patterns to corresponding patterns in a star catalog [18]. The probability analysis of pattern match algorithm is also studied in [43].

This chapter Introduces this probability-based method to characterize the likelihood of an incorrect star identification, and since the expected frequencies derived are general functions of the measurement precision, number of stars the camera can image, and the measured interstar angles, this will hereafter negate the need for expensive Monte Carlo type simulations for each star sensor design variation. The fundamental innovation of this chapter is that analytical expressions are derived for the expected frequency of randomly matching measured star patterns with patterns in the star catalog simply due to the number of cataloged stars and measurement error. This expected frequency of a random invalid match can be used to rigorously terminate the star pattern matching process with an essentially certain star identification.

This analytical method to characterize the frequency of an incorrect star identification, is an innovation of significant theoretical and practical importance, since the expected frequencies derived are general functions of the measurement precision, number of stars the camera can image, and the measured interstar angles. These general formulas will hereafter negate the need for expensive and often inconclusive Monte Carlo simulations for each star sensor design variation. As associated and highly robust *Pyramid* method, described in chapter (IV) is introduced for star pattern identification, for the general lost-in-space case of no prior information.

In any event, we anticipate that over the course of the next decade, there is at least occasional need for star identification algorithms with an overall expected frequency of miss-matches approaching 10^{-10} and for longer missions with high star camera frame rates, we can conceive of a need for exceptionally small expected star miss-match frequencies (perhaps even $< 10^{-20}$). Obviously validating such frequency estimates is compounded because with every sensor design change, and every resetting of any variable system parameter may necessitate repeating the miss-match frequency analysis. It is evident that Monte Carlo processes is impractical in this situation. Like the perpetual thirst for faster computers, we can never develop a star identification algorithm that fails too rarely! Moreover, even without pursuing such small frequencies of spurious star identifications, it is very obvious that having the capability to quantify and minimize the frequency of failure is fundamental to analyzing/optimizing overall mission reliability. Therefore, we expect that the formulas developed in this chapter and future refinements [44] to find a very practical home in sensor design and mission analysis by eliminating the reliance on slowly converging statistical simulations such as Monte Carlo processes.

8.2 Probability Quantification of the Star-ID Process

This section provides the mathematical tools which establish, in a closed form, the reliability of the star identification process associated with the three most fundamental star structures used in the star-ID algorithms. These star structures consider the interstar angles associated polygon sets of p stars such as a pair, or triangle of stars, as well as a pyramid of four or more stars. For more complex star structures, the associated reliability quantification, can easily be derived from the fundamental star structures formulae for the first few cases presented here.

Let us consider the whole sky with a uniform star distribution. This implies that the star density ρ (which depends on the given magnitude threshold m), is simply given as

$$\rho(m) = \frac{N(m)}{4\pi} \tag{8.1}$$

where $N(m)$ is the overall number of stars with magnitude less than m. The relationship between magnitude threshold m and $\ln\{N(m)\}$ can be approximated (see [30]) by the following linear expression

$$m = 0.8985 \, \ln N - 2.0474 \tag{8.2}$$

which is obtained by least square best fitting. Slightly better choices imply higher degree polynomials. For instance, for quadratic best fitting, least square approximation yields

$$m = 0.0126 \, (\ln N)^2 + 0.7109 \, \ln N - 1.3734 \tag{8.3}$$

Figure 8.1 shows residuals between the linear and quadratic best fits together with the associated standard deviations. We conclude that Eq. (8.3) provides an adequate approximation, especially for $N > 1000$, because the star camera determined magnitude seldom matches to better than 0.1 magnitude variations when compared to a

cataloged value. [1]

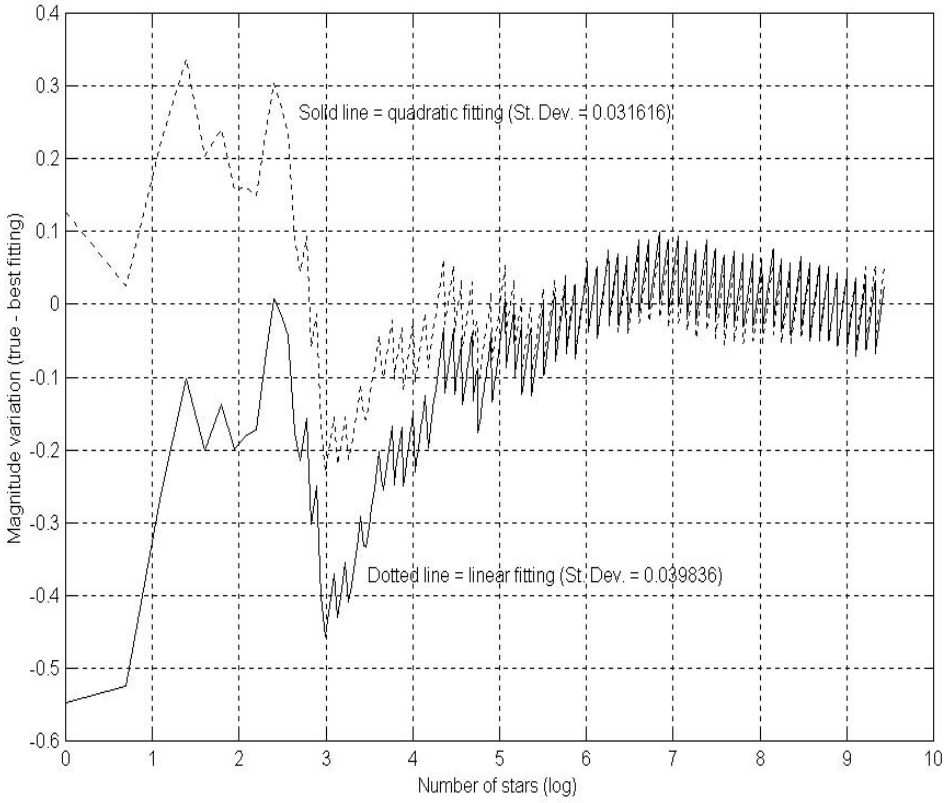

Figure 8.1 Residuals for linear and quadratic best fits

Let us consider the spherical surface defined by a cone of aperture ϑ, that is, the area

$$S(\vartheta) = 2\pi\,(1 - \cos\vartheta) \tag{8.4}$$

and let us consider that the axis of this cone is aligned with the i-th star. Referring to Fig. 8.2, in the infinitesimal spherical area $dS(\vartheta)$, that can be evaluated as the difference between two cones of apertures $(\vartheta + d\vartheta)$ and ϑ, and which has the area

$$dS(\vartheta) = S(\vartheta + d\vartheta) - S(\vartheta) = 2\pi\,\sin\vartheta\,d\vartheta \tag{8.5}$$

[1]Note that the frequency appearing in these residuals depend on the fact that the star catalog provides the magnitude information with precision truncated to 0.1.

the expected number of stars falling in $dS(\vartheta)$ is

$$dn(\vartheta) = \rho^* \, dS(\vartheta) = \frac{(N-1)}{2} \, \sin\vartheta \, d\vartheta \tag{8.6}$$

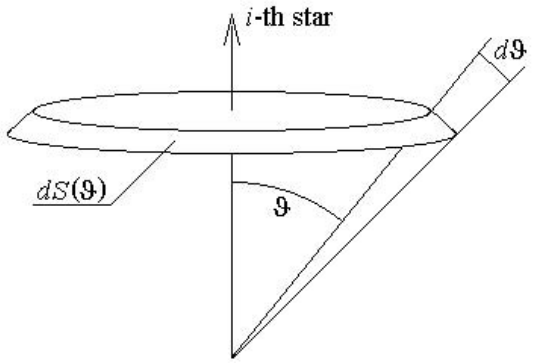

Figure 8.2 Infinitesimal spherical area

where $\rho^* = (N-1)/(4\pi)$ indicates a uniform star density which slightly differs from the expression of the uniform star density ρ given in Eq. (8.1).

This difference is due to the fact that, by aligning the axis of the cone with the i-th star, we are forced to delete that star from the overall number of stars available in the counting of the star density.

Now, the product

$$dn_{ij}(\vartheta) = \frac{1}{2} \, N \, dn(\vartheta) = \frac{N(N-1)}{4} \, \sin\vartheta \, d\vartheta \tag{8.7}$$

provides the number of star pairs "i-j" separated by an angle ranging from ϑ to $(\vartheta + d\vartheta)$. The division by 2 is due to the fact that the product $N \, dn(\vartheta)$ double counts the number of the star pairs, because it considers the star pair "i-j" *and* the same star pair "j-i". By integrating Eq. (8.7) over the whole sky, it is possible to find the number of possible combinations C_N of N objects (stars) taken two by two

$$C_N = \int_0^{\pi} dn_{ij}(\vartheta) \, d\vartheta = \frac{N(N-1)}{2} \tag{8.8}$$

This equation can be obtained from independent considerations, notice that it is the number of combinations of N objects taken two at a time, and thus we have an independent check on Eq. (8.8). Hence, the expectation of the overall number of admissible star pairs n_{ij} displaced by an angle which varies from $(\vartheta_{ij} - k\sigma)$ to $(\vartheta_{ij} + k\sigma)$, is found by integrating Eq. (8.7) over this small region as

$$n_{ij} = \frac{N(N-1)}{4} \int_{\vartheta_{ij}-k\sigma}^{\vartheta_{ij}+k\sigma} \sin\vartheta \, d\vartheta = \frac{N(N-1)}{2} \sin(k\sigma) \sin\vartheta_{ij} \tag{8.9}$$

This equation represents the *expected frequency that false matches between measured "objects", to within measurement precision, are matched by random pattern combinations in the catalog*, assuming a uniform star density.

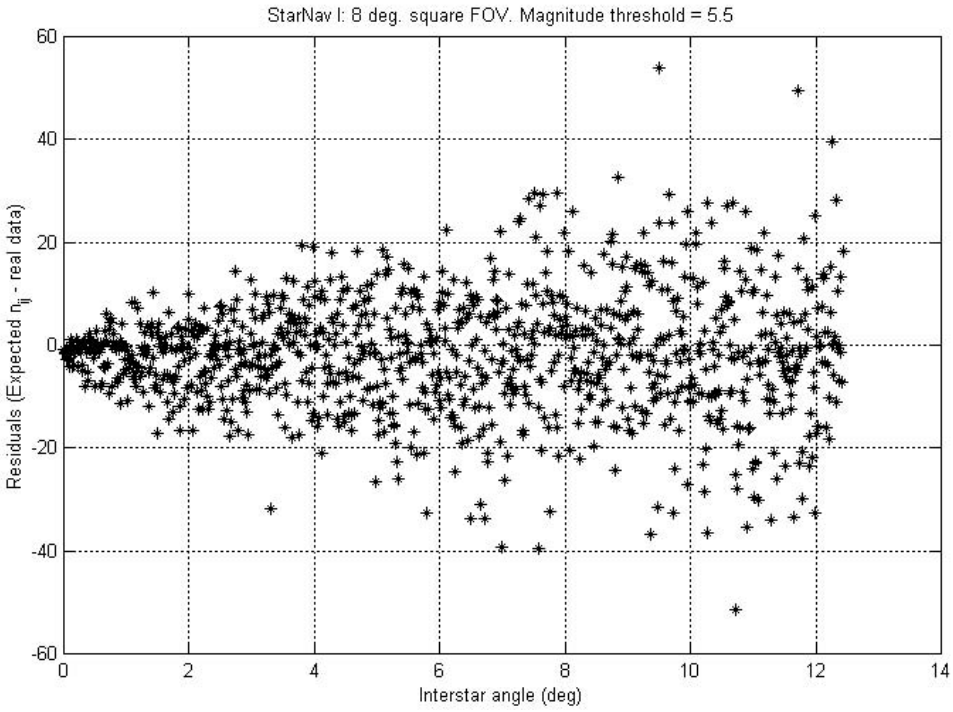

Figure 8.3 Residuals between equation (8.9) and random simulated data

We note that the actual star distribution is not uniform, however, simulations indicate that at most factors of two frequency variations occur, and thus we must

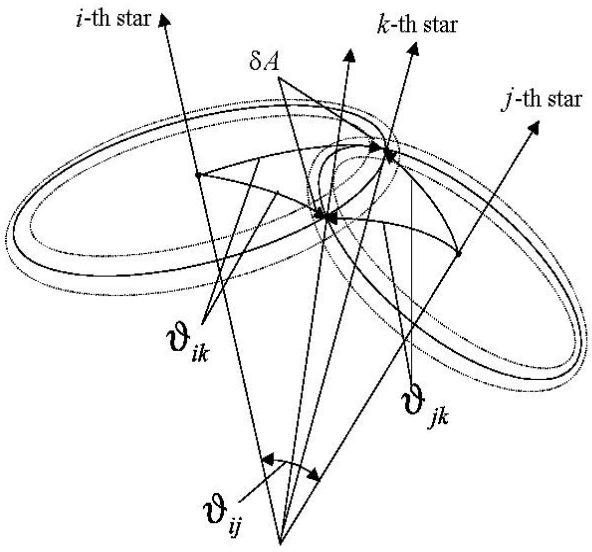

Figure 8.4 Differential area associated with measurement error for three measured stars

take this into account conservatively in interpreting the frequency results obtained (i.e., it would be unwise to believe the frequencies exactly, but we can usually tolerate the factor of two difference between small numbers such as 1×10^{-7} *or* 2×10^{-7}!).

Figure 8.3 shows, for the StarNav I experiment (8 deg square FOV, magnitude threshold = 5.5, 512×512 pixel CCD, focal length = 50 mm, and $3\sigma = 10$ arcsec), the residuals between the values for n_{ij} provided by Eq. (8.9) and random simulated data. After some experimentation, we found that adopted value for k should be about 6.4, a value which is somewhat greater than the 3-σ value of $k = 3\sqrt{2}$ (derived for an interstar angle associated with two stars whose direction precision is normally distributed); this has been found to essentially guarantee that the actual star pair measured is contained in the k-vector subset with n_{ij} elements. Adopting k of about 6.4 ensures that we obtain essentially of all the possible measured stars as candidate stars, even with the actual non-uniform star density. For typical parameter settings (see below), we find $n_{ij} = 200$ is typical, so $\sigma_{n_{ij}} \cong 0.1 n_{ij}$ and apparently the approx-

imation of uniform density leads to moderate errors. However, this approximation is indeed adequate for order of magnitude analysis.

8.2.1 Identification of a Star Pattern with Two Legs

Consider the case of a three star pattern ijk as shown in figure 8.4. We seek to match the measured interstar angles $(\vartheta_{ij} \pm k\sigma)$ and $(\vartheta_{ik} \pm k\sigma)$. To do this, let us consider the i-th star. Now, using Eq. (8.6), it is easy to evaluate the number $\bar{n}_{ij}$ of stars j displaced from i by an angle which varies from $(\vartheta_{ij} - k\sigma)$ to $(\vartheta_{ij} + k\sigma)$. This number is

$$\bar{n}_{ij} = \int_{\vartheta_{ij}-k\sigma}^{\vartheta_{ij}+k\sigma} dn(\vartheta) = (N-1)\,\sin(k\sigma)\,\sin\vartheta_{ij} \tag{8.10}$$

Analogously, the number $\bar{n}_{ik}$ of stars k displaced from i by an angle which varies from $(\vartheta_{ik} - k\sigma)$ to $(\vartheta_{ik} + k\sigma)$ is

$$\bar{n}_{ik} = \int_{\vartheta_{ik}-k\sigma}^{\vartheta_{ik}+k\sigma} dn(\vartheta) = (N-1)\,\sin(k\sigma)\,\sin\vartheta_{ik} \tag{8.11}$$

Hence, the frequency that a star matches with both legs (star pairs ij and ik) is

$$f_{i-(j,k)} = N\,\bar{n}_{ij}\,\bar{n}_{ik} = N\,[(N-1)\,\sin(k\sigma)]^2\,\sin\vartheta_{ij}\,\sin\vartheta_{ik} \tag{8.12}$$

We published these results in [18], along with additional formulas for higher order star polygons. The results for more than three stars need further research and validation.

8.3 The Selection of Star Sensor FOV Size

The size of the Star Tracker field of view is one of the most important aspects in the design of the tracker sensor. The selection of the FOV size depends on the number of star tracker fields of view. For high accuracy on all three axes, it is well known that at least two star tracker fields of view are required.

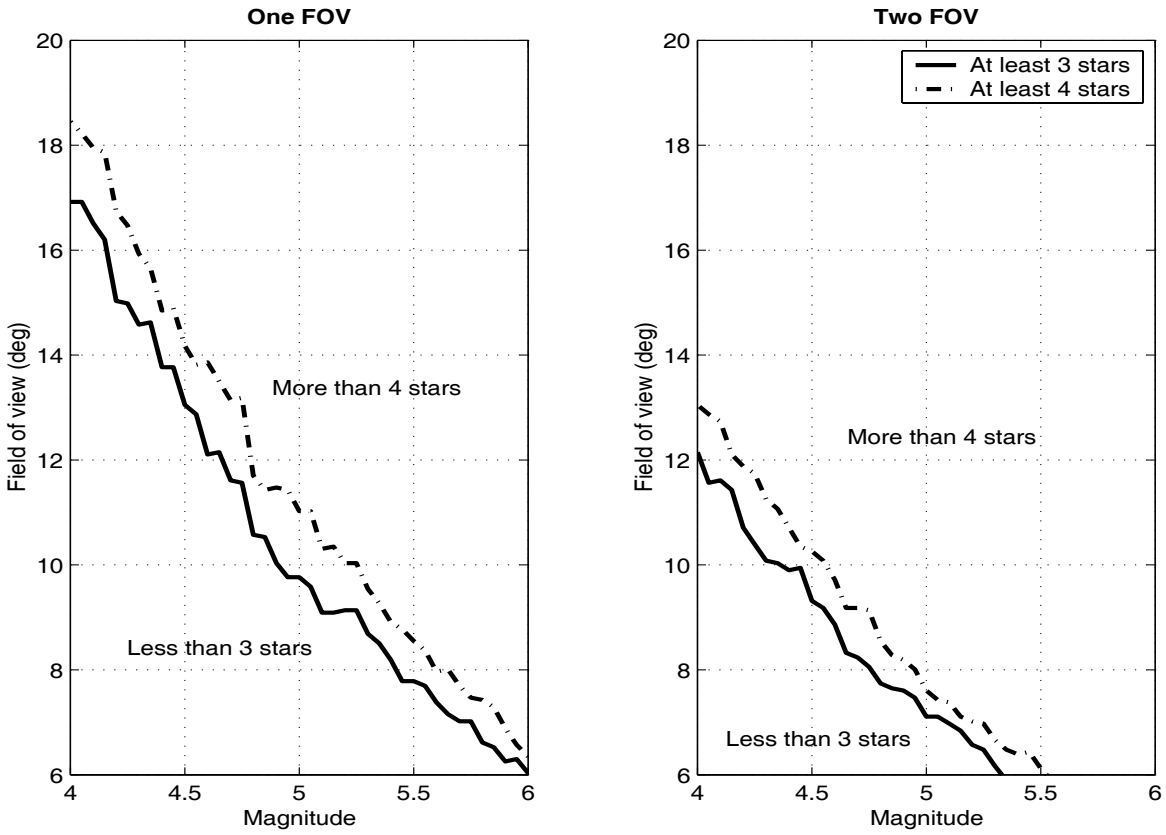

Figure 8.5 The selection of the FOV size

The probability of finding certain number of stars is used to determine the FOV size for each threshold magnitude. While the FOV size analysis can be approached analytically if we assume uniform star density, it is fairly simple to use the actual star catalog and a simple Monte Carlo process. For each magnitude, 1000 tests have been done to obtain the corresponding FOV size that gives the following probabilities

$$P(\text{No. of stars} < 3) < 0.01, \quad \text{and} \quad P(\text{No. of stars} < 4) < 0.01 \tag{8.13}$$

Figure 8.5 shows a plot of the size of square FOV versus the magnitude for magnitude range, (from 4.0 to 6.5), also the same plots are given for two identical star trackers.

CHAPTER IX

CONCLUSIONS

This dissertation has been addressed to developing improved techniques to determine spacecraft attitude using one and two field of view star trackers. All the work in this dissertation has culminated with the completion of an experimental system combining these new techniques. These techniques have been successfully demonstrated in ground-based testing, with real image data, with simulated data, on-orbit experiment like StarNav I on the space shuttle STS-107, and finally the GIFTS mission on the EO-3 spacecraft that will fly on 2005. Many novel techniques for image centroiding, star identification, ground calibration and attitude estimation have been presented in this work.

Chapter II presents predictive centroiding as a new approach for fast image processing for the star trackers. It enables only several hundred pixels to be processed (as opposed to 10^5 or 10^6 pixels); this approach is especially well-suited to Active Pixel cameras which permit random access of the pixel response due to selected stars. The speed and the accuracy of this approach is successfully demonstrated in comparison with the ordinary centroiding algorithms which don't use the previous image data. The predictive centroiding algorithm will be used in the GIFTS EO-3 mission.

Chapter III presents an algorithm to estimate the three focal plane parameters, the focal length and the two principal point offsets using the standard nonlinear least-squares estimation. These parameters are very important for the star identification algorithm because it used to calculate the measured vectors of the imaged stars. Any inaccurate values for these camera parameters may lead to wrong star identification. Ground-based testing using real star images is required to estimate the values of (x_o, y_o

and f). The algorithm presented is based upon measurements of inter-star angles; this measurement is independent of orientation. As a consequence, measurements from an arbitrary number of image frames can be combined to improve the observability.

In chapter IV, we have introduced a novel method for star pattern identification based on matching interstar angles between measured vectors to those from a star catalog. We consider the case of no prior information and our algorithm (*Pyramid* LISA) for solving the Lost In Space problem, is shown to be highly tolerant of spurious events such as reflections off spacecraft debris. *Pyramid* LISA is also believed to be the most efficient algorithm for solving the Lost In Space Problem, even with ~ 6000 cataloged stars, a high percentage of spurious images, and no prior information, we can typically identify a measured star field within a small fraction of a second within the constraint of routinely available computers. Such a speed and robustness have been achieved because the *Pyramid* LISA uses the k-vector method to access all feasible candidate catalog stars *without searching* for any measured pair, and because the pyramid idea is established and supported by an analytical expected random frequencies associated with matching interstar angles from measured star polyhedra. Finally, we mention that the computational and night-sky experimental validations of the results of this chapter will be augmented by an on-orbit validation on the Draper's Inertial Stellar Compass (ISC) for the New Millenium Program, and - more important - for the GIFTS (Geostationary Imaging Fourier Transform Spectrometer) mission in EO-3 spacecraft, which will first validate the concept of multiple fields of view star trackers.

Also, chapter V presents two new recursive methods for identifying the stars within the field of view during camera slewing motion. The spherical polygon approach is used to sort the star position vector components in $\mathbf{x}$, $\mathbf{y}$ and $\mathbf{z}$-axes and then the star identification is done by accessing the cataloged stars within the FOV

corners and the common stars of each sorted vector in **x**, **y** and **z** directions. The star neighbourhood approach is also used as a second method for star identification. In this method we access (without searching) all feasible cataloged star in the union of the neighbourhoods of stars identified in the previous frame. Logic based upon matching the interstar angles between the identified stars and the observed stars provides the final checks.

Chapter VI presents a novel Star-ID method for an Un-calibrated star camera, called *Non-Dimensional Star-ID*. The proposed method is derived from the fact that the angles of focal plane triangles (formed by the star locations), are weakly dependent on the camera focal length, and on the optical axis offsets. The proposed method is particularly suitable for a poorly calibrated camera, and it can be adopted as a highly reliable back-up approach to solve the lost-in-space case of the Star-ID process. Finally, this approach is particularly suitable to make the initial ground calibration for night-sky experiments easier, as well as the in-flight calibrations.

In chapter VII, the image smear problem is studied when we have high S/C angular velocity or high integration time of the camera. We conclude image smear can present significant problems for the astigmatism based optical tagging in split field of view cameras. This problem requires further study. We show simulation results that indicate the actual S/C angular velocity can be increased to 2 or 3 times the value of the critical analytical angular velocity estimate and the centroid measurement errors will be within the allowable limits.

Finally in chapter VIII, we derive the expected random frequencies associated with matching interstar angles from measured star polyhedra. Using recursively the frequency analysis for matching triangles, we show how to develop the expected frequency for four and five star patterns with associated formulas derived for the expected frequency of random matches to within measurement precision. These for-

mulas are original contributions which permit, for the first time, an analytical basis for deciding upon the validity of an identified star pattern. However, the results for more than three stars have not been validated, further research is needed. We also present some results which allow us to consider near double stars in the pattern recognition process, even when the stars are so close together that they cannot be distinctly centroided. We expect that the formulas developed in chapter (VIII) and future refinements [44] to find a very practical home in sensor design and mission analysis by eliminating the reliance on slowly converging statistical simulations such as Monte Carlo processes.

At the end of this thesis, we mention that the computational and night-sky experimental validations of the results of this work were augmented by an on-orbit validation (StarNav I experiment aboard the ill-fated space shuttle Columbia, STS-107) during mid January 2003. Also, most of the algorithms presented in this work are utilized as advanced software designed for StarNav II. StarNav II has been adopted for the GIFTS EO-3, scheduled for a 2007 launch. StarNav II is a new generation autonomous tracker involving several hardware and software innovations:

Split Field of View Optics Two Star Fields separated by 90 are simultaneously imaged by a single detector, the stars are "tagged" optically with unique astigmatism that permits the field of view of the origin for each star to be inferred uniquely during image processing

Active Pixel CMOS Detector Radiation hard, high frame rate, random access of pixel response.

Lost-In-Space Star Identification Using the Pyramid algorithm (described in chapter IV), which is extremely robust and highly efficient, star identification is accomplished in a small fraction of a second.

Recursive On-Orbit Calibration The on-orbit calibration is used for estimating the higher order focal plane distortion for a star tracker camera [21] ,[45].

REFERENCES

[1] Junkins, J. L. and Turner, J. D., *Optimal Spacecraft Rotational Maneuvers*. Elsevier, New York, NY, 1986.

[2] Wertz, J. R., *Spacecraft Attitude Determination and Control*. Reidel, Dordrecht, The Netherlands, 1984.

[3] Shuster, M. D., "A Survey of Attitude Representations," *The Journal of the Astronautical Sciences*, Vol. 41, 1993, pp. 439–517.

[4] Junkins, J. L. and Strikwerda, T. E., "Autonomous Star Sensing and Attitude Estimation," Paper 79-013, *Proceedings of the Annual Rocky Mountain Guidance and Control Conference*, Keystone, CO, February 1979.

[5] Strikwerda, T. E. and Junkins, J. L., "Star Pattern Recognition and Spacecraft Attitude Determination," U.S. Army Engineer Topographic Laboratories, ETL-0260, Fort Belvoir, VA, May 1981.

[6] Cassidy, L. W. and Abreu, R., "Star Trackers for Spacecraft Application," *Acquisition,Tracking, and Pointing IV, Proceedings of SPIE*, Vol. 1304, 1990, pp. 58–74.

[7] Eisenman, A. R., Liebe, C. C., and Joergensen, J. L., "The New Generation of Autonomous Star Trackers," *Sensors, Systems, and Next-Generation Satellites, Proceedings of SPIE*, Vol. 3221, 1997, pp. 524–535.

[8] Samaan, M. A., Mortari, D., Pollock, T. C., and Junkins, J. L., "Predictive Centroiding for Single and Muliple FOVs Star Trackers," *AAS/AIAA Space Flight Mechanics Meeting*, San Antonio, TX, January 2002.

[9] Samaan, M. A., Pollock, T. C., and Junkins, J. L., "Predictive Centroiding for Star Trackers with the Effect of Image Smear," *Journal of the Astronautical Sciences*, Vol. 50, 2002, pp. 113–123.

[10] Junkins, J. L., Pollock, T. C., and Mortari, D., "Multiple Field of View Optical Imaging System and Method," U.S. Patent Pending No. 60/239,559, January 2001.

[11] Cox, J. A., "Evaluation of Peak Location Algorithms with Subpixel Accuracy for Mosaic Focal Planes," *Proceedings of SPIE,*, Vol. 292, 1981, pp. 288–295.

[12] Salomon, P. M. and Glavich, T. A., "Image Signal Processing in Subpixel Accuracy Star Trackers," *Proceedings of SPIE*, Vol. 252, 1980, pp. 64–74.

[13] Fossum, E. R., Bartman, R. K., and Eisenman, A. R., "Application of the Active Pixel Sensor Concept to Guidance and Navigation," *Space Guidance, Control, and Tracking, Proceedings of SPIE*, Vol. 1949, 1993, pp. 256–264.

[14] Yadid-Pecht, O., Clark, C. C., Pain, B., Staller, C. O., and Fossum, E. R., "Wide-dynamic-range APS Star Tracker," *Solid State Sensor Arrays and CCD Cameras, Proceedings of SPIE*, Vol. 2654, 1996, pp. 82–92.

[15] Liebe, C. C., Dennison, E. W., Hancock, B. R., Stirbl, R. C., and Pain, B., "Active Pixel Sensor (APS) Based Star Tracker," *Proceedings of IEEE Aerospace Conference*, Aspen, CO, March 1998.

[16] Shalom, E., Alexander, J. W., and Stanton, R. H., "Acquistion and Tracking Algorithms for the ASTROS Star Tracker," *The Annual Rocky Mountain Guidance and Control Conference*, Keystone, CO, January 1985.

[17] Mortari, D., "Attitude Demonstration Using the Multiple FOVs Star Tracker StarNav III," ASI Contract NI/184/00/0, Final Report, September 2001.

[18] Mortari, D., Junkins, J. L., and Samaan, M. A., "Lost-In-Space Pyramid Algorithm for Robust Star Pattern Recognition," Paper AAS 01-004 *Guidance and Control Conference*, Breckenridge, CO, Jan-Feb 2001.

[19] Samaan, M. A., Mortari, D., and Junkins, J. L., "Recursive Mode Star Identification Algorithms," *Proceedings of the 2002 AAS/AIAA Space Flight Mechanics Meeting*, Santa Barbara, CA, February 2001.

[20] Mortari, D., "Second Estimator of the Optimal Quaternion," *Journal of Guidance, Control, and Dynamics*, Vol. 23, 2000, pp. 885–888.

[21] Samaan, M. A., Griffith, T., Singla, P., and Junkins, J. L., "Autonomous on-Orbit Calibration of Star Trackers," *2001 Core Technologies for Space Systems Conference*, Colorado Springs, CO, November 2001.

[22] Anderson, D., *Autonomous Star Sensing and Pattern Recognition for Spacecraft Attitude Determination*, PhD dissertation, Texas A&M University, College Station, TX, 1991.

[23] Mortari, D. and Neta, B., "k-vector Range Searching Techniques," *The 10th Annual AIAA/AAS Space Flight Mechanics Meeting*, Clearwaters, FL, January 2000.

[24] Samaan, M. A., Mortari, D., and Junkins, J. L., "Non Dimensional Star Identification for Uncalibrated Star Cameras," Paper AAS 03-131 *Space Flight Mechanics Meeting*, Ponce, Puerto Rico, February 2004.

[25] Junkins, J. L., *An Introduction to Optimal Estimation of Dynamical Systems*. Sijthoff and Noordhoff International Publishers B. V., Alphan aan den Rijn, The Netherlands, 1978.

[26] Kowalick, J., *Methods for Unconstrained Optimization Problem*. American Elsevier Publishing Company, New York, NY, 1968.

[27] Crew, G. B., Vanderspek, R., and Doty, J., "HETE Experience with the Pyramid Algorithm," MIT Center for Space Research, Cambridge, MA, June 2002.

[28] Brady, T., Tillier, C., Brown, R., Jimenez, A., and Kourepenis, A., "The Inertial Stellar Compass A New Direction in Spacecraft Attitude Determination," Paper SSC02-II-1, *16th Annual USU Conference on Small Satellites*, August 2002.

[29] Strikwerda, T. E. and Junkins, J. L., "Star Pattern Recognition and Spacecraft Attitude Determination," U.S. Army Engineer Topographic Laboratories, ETL-0260, Fort Belvoir, VA, May 1981.

[30] Mortari, D., "Search-Less Algorithm for Star Pattern Recognition," *Journal of the Astronautical Sciences*, Vol. 45, 1997, pp. 179–194.

[31] Markley, L. F. and Mortari, D., "New Developments in Quaternion Estimation from Vector Observations," *Proceedings of the Richard H. Battin Astrodynamics Symposium Conference*, Texas A&M University, College Station, TX, March 2000.

[32] Wahba, G., "A Least Squares Estimate of Spacecraft Attitude," *SIAM Review*, Vol. 7, 1965, p. 409.

[33] Junkins, J. L., White, C. C., and Turner, J. D., "Star Pattern Recognition for Real Time Attitude Determination," *Journal of the Astronautical Sciences*,

Vol. 25, 1977, pp. 251–270.

[34] Sasaki, T. and Kosaka, M., "A Star Identification Method for Satellite Attitude Determination using Star Sensors," *Proceedings of the 15th International Symposium on Space Technology and Sciences*, Tokyo, Japan, May 1986.

[35] Scholl, M. S., "Star-Field Identification for Autonomous Attitude Determination," *Journal of Guidance, Control and Dynamics*, Vol. 18, 1995, pp. 61–65.

[36] Ju, G., Kim, H., Pollock, T. C., and Junkins, J. L., "Micro Star Tracker and Attitude Determination System," *Third International Symposium on Reducing the Cost of Spacecraft Ground Systems and Operations*, Tainan, Taiwan, March 1999.

[37] Mortari, D., "SP-Search: A New Algorithm for Star Pattern Recognition," *Advances in the Astronautical Sciences*, Vol. 102, 1998, pp. 1165–1174.

[38] Mortari, D. and Junkins, J. L., "SP-Search Star Pattern Recognition for Multiple Fields of View Star Trackers," *Proceedings of the AAS/AIAA Astrodynamics Specialist Conference*, Girdwood, AK, August 1999.

[39] Ju, G., Kim, Y. H., Pollock, T. C., Junkins, J. L., Juang, J. N., and Mortari, D., "Lost-In-Space: A Star Pattern Recognition and Attitude Estimation Approach for the Case of No *a Priori* Attitude Information," *Proceedings of the 2000 AAS Guidance & Control Conference*, Breckenridge, CO, February 2000.

[40] Mortari, D., "ESOQ: A Closed-Form Solution to the Wahba Problem," *Journal of the Astronautical Sciences*, Vol. 45, 1997, pp. 195–204.

[41] Mortari, D., Pollock, T. C., and Junkins, J. L., "Towards the Most Accurate

Attitude Determination System Using Star Trackers," *Advances in the Astronautical Sciences*, Vol. 99, 1997, pp. 839–850.

[42] Mortari, D. and Angelucci, M., "Star Pattern Recognition and Mirror Assembly Misalignment for DIGISTAR II and III Star Sensors," *Advances in the Astronautical Sciences*, Vol. 102, 1998, pp. 1175–1184.

[43] Didinsky, G. and Wu, A., "Probability Analysis of Pattern Match Algorithm for Spacecraft Attitude Acquisition," Paper AAS 98-365 *AIAA/AAS Astrodynamics Specialist Conference*, Boston, MA, Feb 1998.

[44] Mortari, D., Junkins, J. L., and Samaan, M. A., "An Analytical Approach to Star Identification Reliability," in preparation.

[45] Griffith, D. T., Puneet, S., and Junkins, J. L., "Autonomous On-orbit Calibration Approaches for Star Tracker Cameras," *AAS/AIAA Space Flight Mechanics Meeting*, San Antonio, TX, January 2002.

VITA

Malak Anees Samaan was born on April 29, 1973, in Cairo, Egypt. He received his B.Sc. in aerospace engineering from the Faculty of Engineering, Cairo University, Cairo, Egypt in May 1995. Following his bachelor's degree, he started his career work at the National Authority for Remote Sensing and Space Sciences, Cairo, Egypt. He received his M.S. and Ph.D. in aerospace engineering from Texas A&M University, College Station, Texas in May 2000 and August 2003, respectively. While at Texas A&M University, he worked with Dr. John L. Junkins and Dr. Daniele Mortari. His doctoral work focussed on star tracker sensors. He has authored two journal articles and six conference papers. He may be reached through Dr. John L. Junkins at the Aerospace Engineering Department at Texas A&M University, College Station, TX, 77843.

This document was typeset in LaTeX by Malak A. Samaan.